

Títol: Diccionario digital del Benasqués
Volum: 1
Alumne: Lamora Subirá, Santiago
Director/Ponent: Gil Gomez, Maria Luisa
Departament: AC
Data: 22 de septiembre de 2010

DADES DEL PROJECTE

Títol del Projecte: Diccionario digital del Benasqués

Nom de l'Estudiant: Lamora Subirá, Santiago

Titulació: Enginyeria en Informàtica

Crèdits: 37,5 crèdits

Director/Ponent: Gil Gomez, Maria Luisa

Departament: AC

MEMBRES DEL TRIBUNAL (nom i signatura)

Presidenta: Jimenez Castells, Marta

Vocal: Gabarro Vallés, Joaquim

Secretària: Gil Gomez, Maria Luisa

QUALIFICACIÓ

Qualificació numèrica:

Qualificació descriptiva:

Data: 22 de septiembre de 2010

Índice

I	Introducción	1
1.	Motivación	2
2.	Descripción del proyecto	3
3.	Objetivos	4
3.1.	Generales	4
3.2.	Específicos	4
4.	Definición del alcance	5
5.	Organización de la memoria	5
II	Análisis del Proyecto	7
6.	Antecedentes	7
6.1.	Diccionarios digitales publicados	7
6.2.	Proyectos en lengua aragonesa	8
7.	Factibilidad	8
8.	Análisis de requisitos	9
8.1.	Requisitos funcionales	9
8.2.	Requisitos no funcionales	10
9.	Planificación	10
9.1.	Tareas a realizar	10
9.2.	Distribución en el tiempo	11
10.	Estudio económico	13
III	Especificación	14
11.	Digitalización	14
11.1.	Objetivo	14
11.2.	Tareas	14
11.2.1.	Escaneo del diccionario	14
11.2.2.	OCR	15
11.2.3.	ParserOCR	16
11.2.4.	ParserMultilang	16

12. Aplicación Web	18
12.1. Objetivo	18
12.2. Casos de Uso	18
12.2.1. Búsqueda estándar	19
12.2.2. Búsqueda por contexto	19
12.2.3. Palabra aleatoria	19
12.2.4. Login	20
12.2.5. Logout	20
12.2.6. Añadir palabra	20
12.2.7. Modificar palabra	20
12.2.8. Cambio de idioma	21
12.2.9. Mostrar/ocultar información extra	21
 IV Diseño	 22
13. Análisis tecnológico	22
13.1. OCR[22]	22
13.2. Python	22
13.3. Tecnología web y servidor de aplicaciones	23
13.3.1. Tecnología servidor. Java (JSP)	23
13.3.2. Tecnología de la vista. HTML+CSS+AJAX	24
13.4. Sistema de Gestión de Bases de Datos (DBMS)	24
13.4.1. MySQL frente a PostGreSQL	25
13.4.2. MySQL: motores de almacenamiento	25
13.4.3. Bases de datos relacionales frente a Bases de datos XML	26
 14. Arquitectura de la aplicación web	 29
14.1. Patrones de diseño	29
14.1.1. MVC[9]	29
14.1.2. FrontController	29
14.1.3. DispatcherView	31
14.1.4. InterceptingFilter	31
14.2. Diseño de la vista	32
14.3. Diagrama de componentes (diseño interno)	33
14.4. Mapa de navegación	35
14.5. Diseño de la Base de datos[38]	35
14.5.1. Restricciones de Integridad	36
14.5.2. Índices[12]	36
 V Implementación	 39

15. Digitalización	39
15.1. Análisis del material	39
15.2. Alternativas	39
15.3. Digitalización mediante un software OCR	40
15.4. parserOCR / parserMultilang	41
16. Aplicación Web	42
16.1. Aplicación de los patrones arquitectónicos	42
16.2. Implementación de las funcionalidades	43
16.2.1. Búsqueda estándar y multilingüe	43
16.2.2. Palabra aleatoria	43
16.2.3. Palabras cercanas lexicográficamente	43
16.2.4. Búsqueda contextual	44
16.2.5. Login (security): Java Filters	45
16.3. Cuestiones generales de la implementación	46
16.3.1. Internacionalización: Java ResourceBundle[16, 7]	46
16.3.2. Seguridad en aplicaciones web	46
16.3.3. Codificación (encoding, charset, cotejamiento...)	47
VI Conclusiones	49
16.4. Trabajo futuro	50
16.4.1. Webservice	51
16.4.2. Sistema de notificación de errores	51
16.4.3. Diccionario ilustrado	51
16.4.4. Conjugador de verbos	51
16.4.5. Internacionalización del JavaScript	51
16.4.6. Desarrollar un instalador	51
16.4.7. Usuarios y seguridad login	51
16.4.8. SGBD XML nativo + XSLT en la gestión de contenido	52
Nomenclatura	53
Referencias	54
ANEXOS	57
ANEXO I - Análisis Dictionarios Digitales	57
ANEXO II - Manual de usuario	59
ANEXO III - Licencia GPL	65

Parte I

Introducción

Este documento describe el proceso seguido para realizar el Diccionario Digital del Benasqués como Proyecto Final de Carrera de Ingeniería en Informática del plan 2003 de la Facultad de Informática de Barcelona. Su código fuente está disponible en línea¹ bajo la licencia GPL (General Public License) así como el resultado del trabajo realizado².

El benasqués (o patués)[2][32]

El benasqués es una lengua hablada exclusivamente en el valle de Benasque, situado en el Pirineo Oriental Aragonés. Según palabras de José Antonio Saura Rami³: “el benasqués se trata de un sistema lingüístico autóctono fruto de la secular evolución del latín tardío en este lugar”[42].

Debe su idiosincrasia a una ubicación privilegiada que combina su esencia fronteriza, con el paso continuado de caminantes y con su vocación de pueblos y aldeas cerradas sobre sí mismas, arrellanadas en la ribera o trepando por la falda de la montaña.

En la actualidad, los recursos destinados a potenciar esta lengua son muy limitados; las aportaciones suelen ser a título particular o impulsadas por asociaciones culturales sin ánimo de lucro como la Asociación Guayente⁴. A través de dicha asociación, Carmen Castán, escritora y profesora de benasqués, manifiesta su necesidad de una herramienta tecnológica que le sirva como material docente.

La incidencia del benasqués en Internet es reducida, su aparición suele presentarse englobada bajo la denominación de aragonés. Susodicha lengua aglutina al conjunto de variedades dialectales distribuidas desde el Pirineo más occidental, valles de Hecho y Ansó, hasta la zona oriental, en la que se encuentra el valle de Benasque (ver figura 1⁵). Estas muestras están presentes en la Wikipedia en aragonés⁶.

¹Código fuente en GoogleCode: <http://jsp-web-dictionary.googlecode.com>

²Diccionario digital del benasqués: <http://www.guayente.org/diccionario/>

³José Antonio Saura Rami, profesor asociado del Departamento de Lingüística General e Hispánica. Es Licenciado en Filología Clásica (Universidad de Salamanca), Doctor en Filología Hispánica (Universidad de Zaragoza) y cursó estudios de Tercer Ciclo en la Università degli Studi di Bologna bajo los auspicios del programa Erasmus.

⁴Asociación Guayente: <http://www.guayente.org>

⁵Adaptación de la imagen http://es.wikipedia.org/wiki/Archivo:Bariedaz_lingüísticas_d'Aragón.png. Publicada bajo la licencia Creative Commons Genérica de Atribución/Compartir-Igual 2.5 (<http://creativecommons.org/licenses/by-sa/2.5/deed.es>)

⁶Wikipedia en aragonés: <http://an.wikipedia.org>

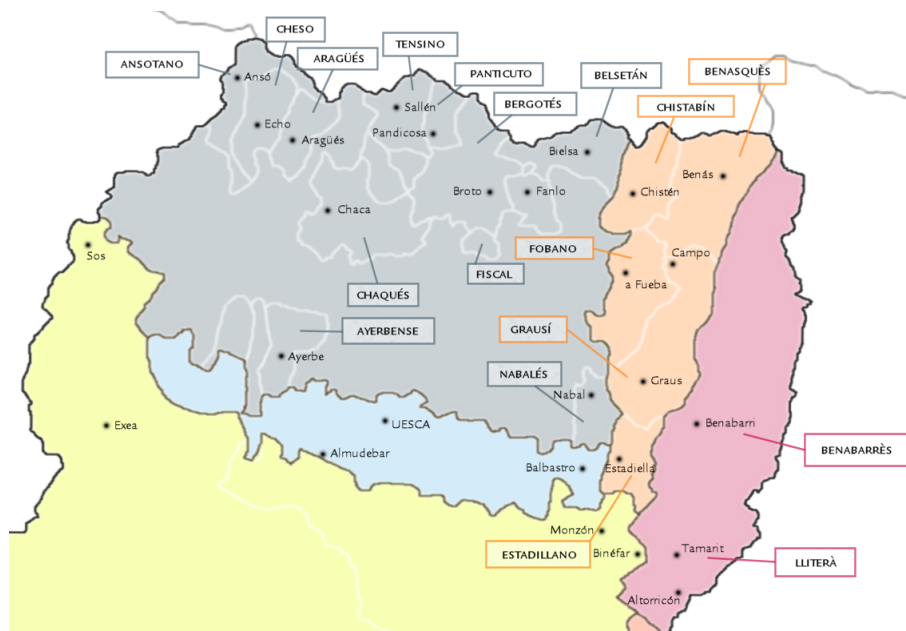


Figura 1: Variedades lingüísticas de Aragón

1. Motivación

El origen de este proyecto se remonta al año 2008 cuando conocí diversos proyectos realizados como herramientas de aprendizaje de lenguas minoritarias[37, 30]. La lingüística era un tema que me interesaba y conocer estos proyectos me inspiró para unir este interés con los conocimientos adquiridos durante la carrera.

El vínculo con la lengua propia de mi región de procedencia, el benasqués, declarada lengua en peligro de extinción se unió con la necesidad manifiesta de una herramienta de soporte a la enseñanza de dicha lengua.

Un diccionario es el instrumento básico para el aprendizaje de una lengua. Desde mi propia experiencia, como estudiante de idiomas, tenía constancia de ello y por eso decidí que ésta era la línea de trabajo que estaba buscando: un diccionario como herramienta de soporte al aprendizaje del benasqués.

Se trata de una herramienta válida a todos los niveles: gente que quiere conocer una lengua, que la está aprendiendo o personas nativas que quieren ampliar su vocabulario o solventar dudas ortográficas o gramaticales.

Además, el diccionario del benasqués, obra de Ángel Ballarín, es mucho más que una colección de palabras. Junto a la definición de las mismas acompaña sus usos más frecuentes en el valle de Benasque, lo cual refleja un modo de pensar, una manera de hacer, pero sobre todo un modo de vida en un entorno hostil pero a la vez bello, como eran los valles pirenaicos. Y como hablamos de una

lengua cuyo vigor está menguando progresivamente, contribuimos a salvar un legado cultural y patrimonial de gran importancia e interés.

Desde una visión práctica, se me ofrecía la oportunidad de aplicar los conocimientos adquiridos durante la ingeniería en un proyecto sociocultural. Además de desarrollar un proyecto cuya vida útil se prolongase más allá de la duración del PFC; el diccionario perdurará como entidad y sus contenidos pueden servir de base para generar nuevos recursos lingüísticos (por ejemplo, actividades de vocabulario).

En conjunción a lo anterior, nos encontramos en la denominada era digital en la que Internet proporciona unos medios para llegar a un amplio número de personas con interés. Por lo tanto, esta herramienta educativa poseerá un alto grado de disponibilidad y accesibilidad.

Por último, comentar que se trata de un proyecto de aplicación local con un enfoque global. La situación del benasqués no es un caso aislado, existen cientos (si no miles) de lenguas minoritarias (y minorizadas)[40, 1] que pueden aprovechar el soporte tecnológico desarrollado en este proyecto. En concreto, cuando iniciamos las investigaciones sobre software existente mantuvimos contacto con una persona de India que estaba iniciando un proyecto similar⁷.

2. Descripción del proyecto

El proyecto consiste en una versión digital de un diccionario en papel; la obra de Ángel Ballarín Cornel, *Diccionario del benasqués*[29].

Se trata de una herramienta lingüística de similares características a otros diccionarios digitales de otras lenguas de la península como, por ejemplo, el diccionario de la Real Academia Española[14] o el desarrollado por la *Enciclopèdia Catalana*, *SAU*[4] para el catalán.

El proyecto se compone de dos partes que pueden ser tratados de forma independiente durante el desarrollo:

1. Digitalizar el contenido: escanear + aplicar *OCR*⁸ + extraer la información (parser) [+ corregir]. [CONTENIDO]
2. Desarrollar la plataforma que permita acceder al contenido (aplicación web). [CONTINENTE]

⁷Web Dictionary at SourceForge: <http://sourceforge.net/projects/web-dictionary/>

⁸OCR (Optical Character Recognition) es la traducción electrónica de imágenes escaneadas que contienen texto escrito en texto codificado en formato de máquina (p.ej.: un fichero de texto).

3. Objetivos

3.1. Generales

1. Crear una herramienta de aprendizaje del benasqués que además potencie su transmisión intergeneracional. Debe recordarse que el benasqués se trata de una lengua, fundamentalmente, de tradición oral. Un contenido digitalizado es un gran método para propagar el conocimiento de esta lengua en un entorno más o menos cercano.
2. Desarrollar una herramienta que pueda ser utilizada para la divulgación de otras lenguas minoritarias. La abstracción realizada en el proceso de la ingeniería del software permite adaptar una aplicación creada con un fin concreto a situaciones más genéricas. En este caso otras lenguas.
3. Publicar el software bajo una licencia que permita a otras personas colaborar en su desarrollo, mejora y difusión del sistema. Abrir al público una herramienta de este tipo significa allanar el camino a las aportaciones de otras personas con objetivos y/o necesidades afines. Considerando que hay más de tres mil lenguas en el mundo, con toda seguridad, el trabajo hecho puede servir a otro colectivo.

3.2. Específicos

1. Proporcionar una versión digital del diccionario de benasqués al dominio público.
2. Permitir el mantenimiento y mejora de los contenidos del diccionario digital. La lengua vive una evolución y cambio continuo, es dinámica; el diccionario como herramienta lingüística debe proporcionar métodos para poder actualizar una información indiscutiblemente variable.
3. Desarrollar una aplicación web que cumpla los estándares. La única manera de garantizar una correcta visualización de los contenidos por el mayor espectro posible de usuarios es basar la implementación según los estándares definidos; además de que es nuestra manera de apoyar el uso de éstos y contribuir a que cada vez más los desarrolladores de software los tengan presentes.
4. Construir una aplicación con soporte a la internacionalización para que pueda ser traducida a otras lenguas. De cara a publicar una aplicación en Internet es imprescindible disponer de su versión en inglés; dado el doble objetivo del proyecto también debe considerarse que los destinatarios de un diccionario de benasqués pueden ser hablantes de benasqués, aragonés, catalán y/o castellano (por proximidad geográfica-lingüística).
5. Diseñar una interfaz de usuario minimalista e intuitiva que basten unos conocimientos mínimos en tecnología para utilizar.

4. Definición del alcance

Por una parte, el proyecto cumplirá el objetivo de generar una versión digital de la información recogida en la obra de Ángel Ballarín. Se intentará realizar un ajuste de las herramientas utilizadas con el fin de reducir los errores en el resultado final pero la revisión manual exhaustiva queda fuera de los límites del proyecto (dado su elevado coste de recursos y, fundamentalmente, de tiempo).

Y por otro lado, en cuanto al desarrollo de la aplicación se fija como límite el conseguir un sistema potencialmente funcional. En otras palabras, se cumplirán las funcionalidades especificadas en este documento realizando unas pruebas básicas de funcionamiento. Sin embargo, como toda aplicación software, no quedará exenta de errores y problemas posiblemente detectados a posteriori. Con lo cual delimitamos el proyecto a generar unos cimientos sólidos de una aplicación que, posteriormente, se podrá ir ajustando al uso y a las necesidades de un contexto concreto.

Podemos describir el sistema como una herramienta de consulta lingüística digital publicada en el entorno web. Su objetivo no es ser una herramienta de traducción, sin embargo proporcionará herramientas orientadas a dar soporte a dicho proceso. La información sobre definiciones que proporciona se basa en los contenidos bibliográficos de la obra de Ángel Ballarín, es decir, palabras en benasqués con su definición en castellano. Por lo que este proyecto no incluye un estudio lingüístico propio.

5. Organización de la memoria

Este documento se organiza en seis partes (incluyendo la actual), seguidas de la bibliografía y los anexos. Cada parte se corresponde con una fase en el ciclo clásico de desarrollo de software:

1. Introducción (la sección actual), relativa al contexto, descripción de objetivos y alcance del proyecto.
2. Análisis, fase de estudio del contexto de aplicación del sistema (problema o carencias a resolver) y de otras herramientas existentes.
3. Especificación, descripción detallada de los criterios de diseño, explicitando el conjunto de requisitos que debe satisfacer el producto.
4. Diseño, etapa de elección de tecnologías concretas y adaptación del modelo teórico a uno dependiente de éstas.
5. Implementación, fase de codificación del proyecto creando un prototipo funcional.
6. Evaluación de resultados y conclusiones, análisis del trabajo realizado valorando el cumplimiento de objetivos fijados en las primeras fases y elaboración de las conclusiones del proyecto.

A lo largo del documento se han sintetizado las ideas más importantes centrándose en los aspectos propios del proyecto. Además se incluyen referencias a cuestiones relacionadas que pueden ampliarse a través de las citas bibliográficas.

Parte II

Análisis del Proyecto

6. Antecedentes

En esta sección realizaremos un breve estudio sobre las herramientas existentes en el ámbito del proyecto: diccionarios digitales (como sistema software) y trabajos en las distintas variantes del aragonés que han tenido como objetivo potenciar la difusión de la lengua.

En primer lugar, aclararemos que los diccionarios digitales son una herramienta que lleva décadas en el mercado. Desde los distribuidos en medios físicos (disquetes, CDs...) hasta los publicados en Internet. Son estos últimos los que mayor interés nos despiertan porque aprovechan el potencial difusor de la red de redes; un potencial a un coste muy reducido en comparación con otros medios tradicionales.

6.1. Diccionarios digitales publicados

Como hemos mencionado, existen multitud de sitios web donde se ofrece un diccionario digital. Por proximidad lingüística y nuestra experiencia previa como usuarios, nuestras referencias han sido el diccionario de la lengua española de la RAE (Real Academia Española) y el *Gran Diccionari de la llengua catalana* publicado por Enciclopèdia Catalana, SAU.

Dichas herramientas existen y son funcionales, pero se tratan de aplicaciones de código cerrado; tras intentar contactar con las respectivas entidades no hemos recibido respuesta alguna. Por lo tanto, claramente, quedan descartadas como punto de partida, refiriéndonos a la fase de diseño y codificación, dado que sólo disponemos de la versión “binaria”.

Llegado a este punto, nuestra opción pasa por realizar un análisis basado en la metodología de la ingeniería inversa⁹, en el que incluimos otras herramientas además de los diccionarios nombrados. Podemos resumir en tres puntos las alternativas analizadas indicando cuál de sus características tomamos como referencia (se incluye el estudio detallado como documentación anexa16.4.8):

1. La Wikipedia[21] dispone de una funcionalidad que muestra un artículo aleatorio, esto nos inspiró para crear un método similar que dé a conocer el benasqués.
2. Wordreference[23] y el Diccionari Català Multilingüe[3] fueron nuestra referencia para aprovechar el estudio comparativo realizado por A. Ballarín (traducciones a otras lenguas).

⁹El objetivo de la ingeniería inversa es obtener información a partir de un producto accesible al público, con el fin de determinar de qué está hecho, qué lo hace funcionar y cómo fue fabricado.

3. El diccionario de la RAE[14] y la Enciclopèdia SAU[4] han sido modelos de referencia en cuanto a la relación entre búsquedas-resultados y las funcionalidades ofrecidas.

6.2. Proyectos en lengua aragonesa

En el área de trabajos publicados en lengua aragonesa, existe un proyecto de digitalizar un diccionario bilingüe¹⁰ aragonés¹¹-castellano, pero discontinuado por su autor. Además no ha publicado ningún tipo de documentación ni el código fuente de la plataforma.

Por otra parte, el trabajo más relevante para el proyecto, por tratarse de un trabajo publicado en benasqués, es un diccionario subvencionado desde Diputación General de Aragón que recoge varios centenares de vocablos distribuidos en formato *PDF* y accesibles a través de Internet¹². Por su forma de distribución carece de interés en el sentido puramente técnico aunque es de una gran valía lingüística.

Centrándonos en el contenido del diccionario, a través de centros de estudios especializados como el Instituto de Estudios Altoaragoneses¹³ y la Facultad de Filología de la UB¹⁴, dimos con dos obras que cumplían los criterios de búsqueda, ser diccionarios en benasqués:

- Vicente Ferraz y Castán. Vocabulario del dialecto que se habla en la Alta Ribagorza (Huesca). Tipografía de la Revista de Archivos, Madrid, 1934.
- Ángel Ballarín Cornel. Diccionario del benasqués. La Editorial, Zaragoza, 2ª edición, 1978.

Durante el proceso de selección se optó por la obra más reciente, alternativa justificada por tratarse de un trabajo más exhaustivo (por su número de vocablos) y que en su contenido recoge ejemplos de uso que reflejan la idiosincrasia de una zona tan peculiar como el valle de Benasque.

Por otro lado, Ballarín, en su labor de estudio de la lengua benasquesa, su lengua materna, incluyó las traducciones a las lenguas vecinas (francés, catalán, aragonés y español). Durante la fase de extracción del contenido se aprovechará este trabajo para desarrollar la funcionalidad de diccionario multilingüe (ver sección 16.2.1).

7. Factibilidad

En esta sección se evalúa la viabilidad [técnica] del proyecto.

¹⁰Dizionario Aragonés-Castellano: <http://www.clannac.com/dizionario/>

¹¹Se trata de la variante conocida como fabla aragonesa, una de las fases de normativización llevada a cabo por Rafael Andolz [28]

¹²Diccionari del patués: <http://www.aragob.es/edycul/patrimo/etno/patues/entrada.htm>

¹³Instituto de Estudios Altoaragoneses (Diputación de Huesca) — <http://www.iea.es>

¹⁴Universidad de Barcelona / Facultad de Filología — <http://www.ub.edu/filologia/>

Durante el apartado anterior, en el cual se presentaba un estudio sobre los trabajos existentes en este área, se concluía que el desarrollo de un sistema software de este tipo es técnica y tecnológicamente posible. Daban fe de ello los sistemas tomados como modelos de referencia.

En cuanto a los aspectos legales, es decir, los derechos de reproducción de la obra, se ha conseguido la cesión a través de la Asociación Guayente. Esto nos permitirá usarla como base para el contenido del diccionario digital.

Dada la antigüedad de la obra, data de 1978, descartamos la posibilidad de conseguir una versión digital. Por lo tanto nuestro primer paso será digitalizar su contenido.

8. Análisis de requisitos

En la presente sección se describen los requisitos funcionales y no funcionales que debe cumplir el sistema.

8.1. Requisitos funcionales

1. Interfaz de búsqueda de la definición de una palabra.
 - a) Cada entrada poseerá el siguiente conjunto de campos:
 - 1) Término a definir (la palabra).
 - 2) Categorización morfológica: nombre, verbo, adjetivo...
 - 3) Definición(es) (incluye frases hechas y ejemplos de uso).
 - 4) Traducción al aragonés, catalán, castellano y francés (pueden existir o no según el contenido de la obra de Ángel Ballarín).
 - b) Existirán varios tipos de búsqueda:
 - 1) La búsqueda se realizará por coincidencia exacta. En caso de no encontrar la palabra en el diccionario, se mostrarán las palabras lexicográficamente más próximas.
 - Término en benasqués (por defecto)
 - Término en aragonés
 - Término en catalán
 - Término en castellano
 - Término en francés
 - 2) La búsqueda por contexto. Se buscarán coincidencias en la definición.
 - c) Definición aleatoria: se muestra una palabra al azar.
2. Interfaz de introducción, modificación y eliminación de entradas:
 - a) Se permitirá la introducción de nuevos vocablos en el diccionario.

- b)* Se ofrecerá un método que permita realizar rectificaciones, correcciones, definición de nuevas acepciones y usos, etc. en las entradas existentes.
- c)* El acceso a la funcionalidad será restringido (usuarios autenticados).

8.2. Requisitos no funcionales

1. Basarse sobre filosofía software libre. Publicar bajo licencia GPLv3
2. Documentación de la fase de desarrollo que permita a otras personas contribuir y mejorar el proyecto.
3. Definir un camino que sea un referente para proyectos similares de otras lenguas, generalmente minoritarias/minorizadas.
4. Específicos de la aplicación web:
 - a)* Cumplir con los estándares web (W3C)
 - b)* Diseño intuitivo, facilitar la tarea al usuario.
 - c)* Simplicidad, no recargado (sobrio, minimalista).
 - d)* Multilingüe (poseer capacidad de internacionalización, “traducible”).
 - e)* Fácilmente adaptable a otros diccionarios (modular).

9. Planificación

Esta sección describe la previsión temporal y la gestión de los recursos utilizados durante el transcurso de las etapas del proyecto.

9.1. Tareas a realizar

Tal y como se aclara en la descripción del proyecto, posee dos partes independientes (ver figura 2) por lo que ambos trabajos puedan realizarse de manera prácticamente simultánea. La excepción es la fase en la que se estructura el contenido para que la aplicación web lo gestione.

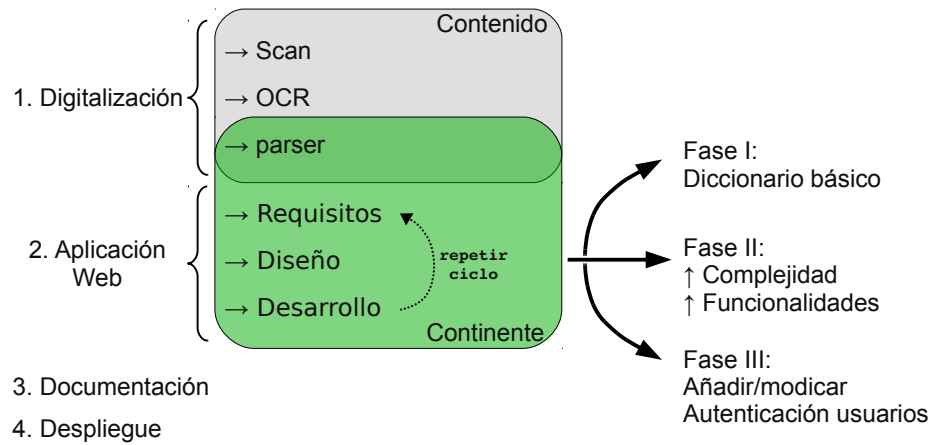


Figura 2: Resumen esquemático de las tareas del proyecto

9.2. Distribución en el tiempo

Antes de describir la planificación del proyecto debe aclararse su origen. En primer lugar, mencionar que el proyecto comienza a desarrollarse en un ámbito ajeno al de un proyecto final de carrera. Por tanto, en su concepción inicial la distribución temporal era distinto.

Tras la adopción del proyecto como PFC, realizamos un trabajo de adaptación, tanto en la metodología como en la planificación (reflejada en la figura 3).

Las tareas desarrolladas con anterioridad a la planificación también se reflejan (la fase de digitalización). Se ha tomado la temporización previa y se ha ajustado agrupando las tareas realizadas en bloques.

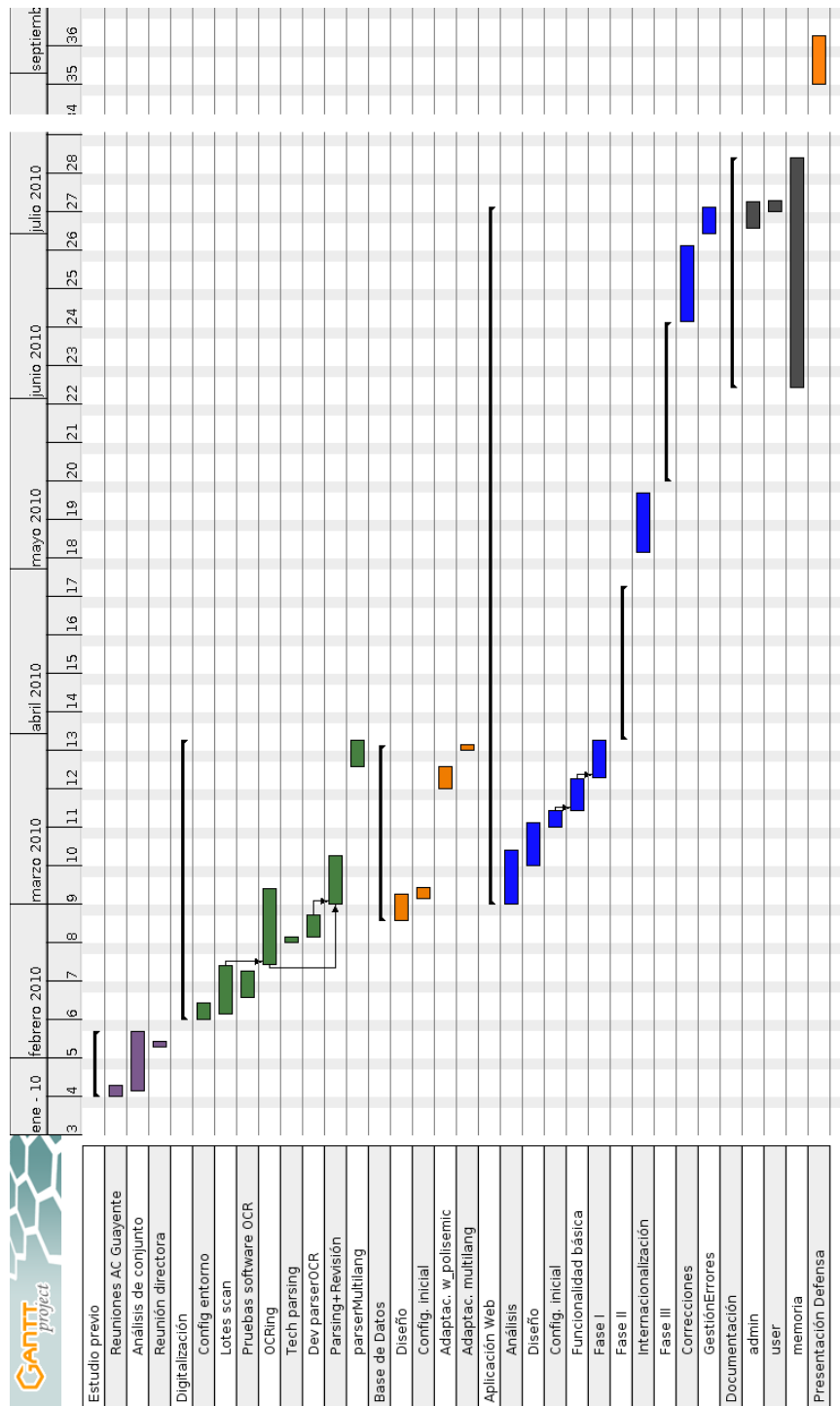


Figura 3: Diagrama de Gantt

10. Estudio económico

Esta sección incluye un análisis de costes económicos derivados de la realización del proyecto. Agrupamos los gastos según sean costes de recursos humanos o materiales.

El importe detallado de los recursos humanos se adjunta en la figura 4. Basándonos en la planificación hemos realizado un cómputo de las horas asociadas a cada perfil (analista, programador, grabador de datos...) y hemos tomado como precio unitario valores reales del mercado laboral.

	Tiempo (horas)			Precio/hora	Coste (€)
	Digitalización	Aplicación Web	[Proyecto]		
Jefe de proyecto	2	8	10	50	500
Analista	8	64	72	40	2880
Diseñador	-	72	72	40	2880
Programador	32	424	456	25	11400
Traductor	-	15	15	15	225
Tester	-	48	48	15	720
Grabador	120	-	120	10	1200
TOTAL	162	631	793	-	19805

Figura 4: Coste recursos humanos

Los costes materiales del proyecto (detallados en la figura 5) incluyen el software y el hardware utilizados para el desarrollo del proyecto.

En el caso del software no ha habido coste alguno por haberse utilizado un entorno de trabajo compuesto al 100 % por herramientas publicadas bajo licencias de software libre (GPL, Apache y BSD).

En cuanto al hardware incluimos el escáner utilizado en la fase de digitalización y el coste de compra del ejemplar en papel del diccionario digitalizado.

	Coste (€)
HP Scan (incluye Readiris 9 Pro)	120
Diccionario del Benasqués	45
Hardware	165
Apache	0
Tomcat 6.0	0
MySQL 5.0	0
NetBeans 6.5	0
GanttProject	0
StarUML	0
OpenOffice 3.2	0
Software	0
TOTAL	165

Figura 5: Coste material

Parte III

Especificación

Esta sección incluye la descripción detallada de los criterios de diseño seguidos en las fases posteriores de la ejecución del proyecto. Como puede apreciarse en la sección 9.1 el proyecto se compone de dos componentes que pueden estudiarse de manera independiente: la digitalización — que se corresponde con el contenido — y la aplicación web — que equivale al continente.

11. Digitalización

11.1. Objetivo

Nuestro fin consiste en transformar la información, actualmente sólo inteligible por un humano (human-readable), en datos procesables por un ordenador (machine-readable).

El trabajo de digitalización es la razón de ser del proyecto.

11.2. Tareas

En la figura 6 se muestran las tareas que lo componen y a continuación acompañamos una breve descripción:

Scan: escaneo de las páginas del diccionario.

OCR: extracción del texto de las imágenes.

parserOCR: estructuración de la información.

parserMultilang: extracción de la información multilingüe.

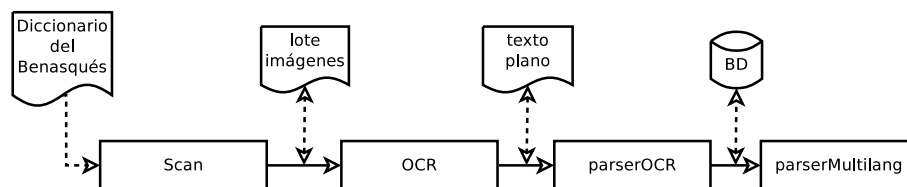


Figura 6: Diagrama de flujo de la fase de digitalización

11.2.1. Escaneo del diccionario

El objetivo de esta tarea es obtener una imagen de cada página del diccionario. En la figura 7 puede verse una muestra del resultado.

Recursos:

- Escáner HP que posee una bandeja de carga de un máximo de 50 páginas.
- hpscan: programa para la obtención de imágenes desde la línea de comandos.
- 505 páginas del diccionario

Especificación: Un sencillo script en Bash automatiza el proceso llamando a hpscan con los parámetros adecuados.

Entrada: lotes de 50 páginas

Salida: 50 imágenes *PNG* con resolución *300dpi*

Tiempo: estimado: 25segundos/página + 1minuto/insertar nueva carga

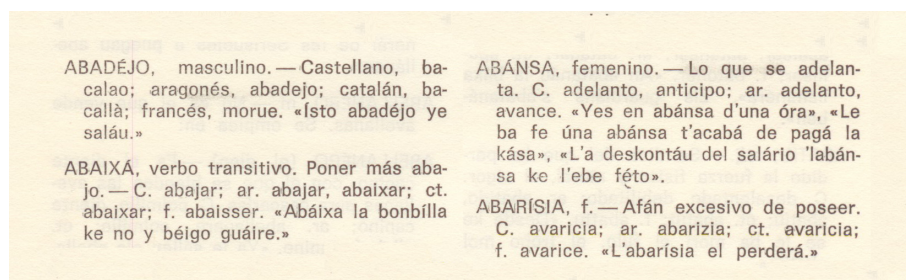


Figura 7: Muestra (parcial) de una página escaneada

11.2.2. OCR

El fin de este proceso es transformar cada imagen escaneada en texto plano.

Recursos:

- 505 imágenes *PNG* a *300dpi*
- Software *OCR*

Especificación

Entrada: lotes de 100 imágenes (limitación del software)

Salida: fichero de texto plano codificado en UTF-8

Tiempo: 75s/imagen

11.2.3. ParserOCR

Se trata de un programa encargado de extraer la información RAW resultante del OCR para transformarla en información estructurada. Dicha información se almacena en la base de datos que posteriormente utilizará la aplicación web.

Especificación Analizamos la estructura de las entradas del diccionario en busca de un patrón que permita desglosar la palabra, su morfología y su definición. El modelo sigue un patrón basado en el formato (mayúsculas, minúsculas) y la puntuación (guiones, comas, etc) descrito por la siguiente expresión regular:

Regular Expression: [WORD] + ',' + [morphology] + ' . -'

pattern = '[A-ZÁÉÍÓÚ] + , [[\wáéíóúñÑ] + \.]*[] -'

Entrada: fichero RAW resultante del OCR.

Salida: BD MySQL; tablas word, definition...

Pre:

1. Separador de entradas = \n (salto de línea)
2. Estructura entrada = TÉRMINO, morfología.- definición

Post:

1. Cada entrada se inserta desglosada en los campos de la tabla 'words' cuyas columnas son [id, term, morf, definition]
2. Se unifica el formato de la morfología {m.|f.|v.|adj.|adv.|*}

11.2.4. ParserMultilang

Se trata de un script que analiza el contenido de la definición de cada palabra (word.definition) buscando el patrón que sigue la información multilingüe. Dicha información, que es la traducción de la palabra al aragonés, catalán, español y francés, la almacena en la base de datos..

Entrada: (base de datos) tabla que contiene las definiciones (word).

Salida: (base de datos) tabla que contiene la información multilingüe (multilang).

Pre:

1. Cada definición puede contener (pero no necesariamente) la traducción a aragonés, catalán, español y francés.

2. La definición sigue el siguiente patrón:
\$lengua + PUNTO + \$contenido + PUNTO | PUNTO_Y_COMA
Ejemplo: [...] fr. garçon; ar. chico. [...]

Post: Se ha insertado en la tabla multilang las traducciones a aragonés, catalán, español y francés (en el caso de que existiesen).

12. Aplicación Web

La sección actual describe el proceso seguido para construir la aplicación web del diccionario.

12.1. Objetivo

El objetivo de este componente es proporcionar una interfaz de acceso a la información digitalizada. La elección de una aplicación web viene dada por ser un modelo que proporciona acceso desde cualquier sitio y a cualquier persona sin la necesidad de un cliente software específico (basta con un navegador web).

12.2. Casos de Uso

Tras el análisis realizado en la sección 6.1 definimos el comportamiento deseado para el sistema.

A continuación se describen los casos de uso principales con el fin de sintetizar los requisitos potenciales de la aplicación web. En la figura 8 puede verse el diagrama de interacción de los stakeholders con la aplicación.

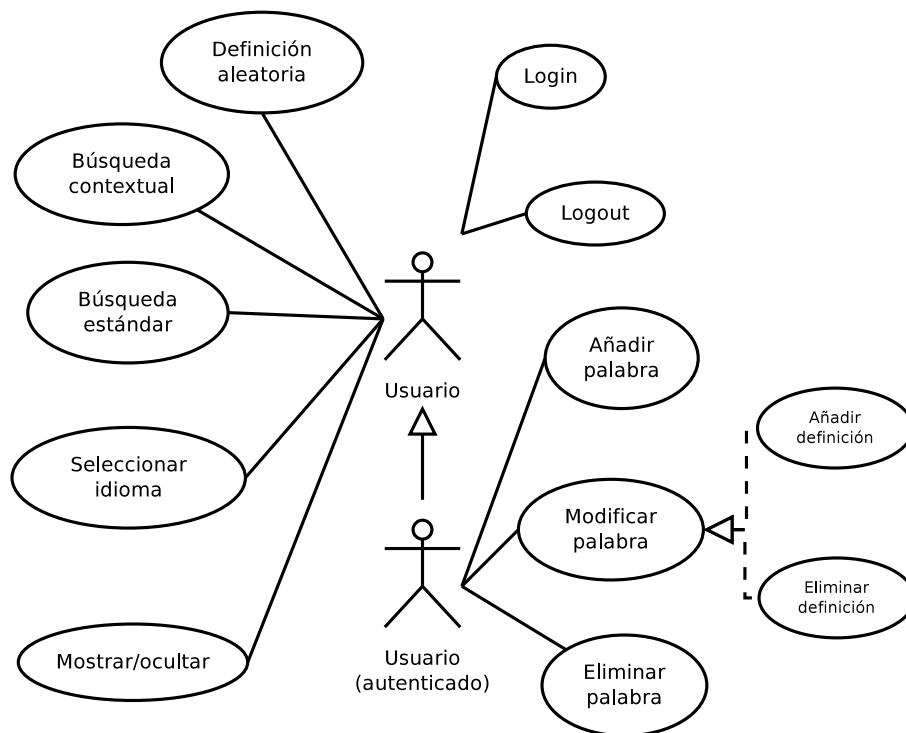


Figura 8: Diagrama de Casos de Uso de la Aplicación Web

12.2.1. Búsqueda estándar

1. El sistema muestra el formulario de búsqueda.
2. El usuario introduce la palabra a buscar.
3. El usuario valida la entrada eligiendo [ar|be|ca|es|fr].
4. El sistema muestra:
 - a) La(s) definición(es) coincidente(s)
 - b) Un listado de palabras alfabéticamente cercanas en [ar|be|ca|es|fr]
5. Error: fallo de conexión a la base de datos (informa al usuario).

12.2.2. Búsqueda por contexto

1. El sistema muestra el formulario de búsqueda.
2. El usuario introduce la(s) palabra(s) a buscar.
3. El usuario valida la entrada.
4. El sistema:
 - a) Muestra la(s) palabra(s) que contienen la frase en su definición, resaltando las apariciones.
 - 1) La definición es coincidente si contiene la(s) palabra(s) completa(s).
 - 2) El resultado sigue el comportamiento de búsquedas literales, es decir, si se introduce la frase “donáse boyá” devuelve las definiciones que contienen ambas palabras ubicadas contiguamente (y en ese orden).
 - b) Informa de que no se han encontrado entradas coincidentes.
 - Sigue el comportamiento de búsquedas literales
 - La definición es coincidente si contiene la(s) palabra(s) completa(s).

12.2.3. Palabra aleatoria

1. El sistema muestra la interfaz.
2. El usuario selecciona la opción de palabra aleatoria.
3. El sistema muestra una definición aleatoria.

12.2.4. Login

1. El sistema muestra el formulario de login.
2. El usuario introduce la información de autenticación.
3. El sistema:
 - a) Muestra un mensaje de bienvenida y permite el acceso a las funcionalidades restringidas (añadir y modificar palabra).
 - b) Informa de que la combinación usuario-contraseña es incorrecta.

12.2.5. Logout

[user logged]

1. El usuario activa el mecanismo de salida del sistema.
2. El sistema muestra el formulario de login.

12.2.6. Añadir palabra

[user logged]

1. El sistema muestra el formulario.
2. El usuario introduce la información de la nueva palabra.
3. El sistema valida la entrada:
 - a) Informa de que se ha insertado correctamente la nueva palabra.
 - b) Muestra un mensaje de error informando del problema. Vuelve al paso (2).

12.2.7. Modificar palabra

[user logged]

1. El sistema muestra el formulario completando los campos con la información actual de la palabra.
2. El usuario actualiza la información de la palabra.
3. El sistema valida la entrada:
 - a) Informa de que se ha actualizado correctamente la nueva palabra.
 - b) Muestra un mensaje de error informando del problema. Vuelve al paso (2).

12.2.8. Cambio de idioma

1. El sistema muestra un listado con los idiomas disponibles.
2. El usuario selecciona uno de ellos.
3. El sistema muestra la página en el idioma seleccionado.

12.2.9. Mostrar/ocultar información extra

Bajo petición del usuario se mostrará información auxiliar como: la ayuda contextual, la licencia de la aplicación, la dirección de contacto o el manifiesto del benasqués.

1. El sistema muestra cualquier sección de la aplicación web.
2. El usuario acciona el mecanismo para mostrar información auxiliar.
3. El sistema muestra u oculta la información auxiliar asociada.

Parte IV

Diseño

13. Análisis tecnológico

En esta sección presentaremos brevemente las tecnologías escogidas para el desarrollo del proyecto justificando su elección frente a otras alternativas consideradas.

13.1. OCR[22]

La tecnología de Reconocimiento Óptico de Caracteres (Optical Character Recognition) es la traducción electrónica o mecánica de imágenes escaneadas que contienen texto escrito (a mano, mecanografiado o impreso) en texto codificado en formato de máquina (p.ej.: un fichero de texto).

Se usa con frecuencia para convertir libros y documentos en ficheros electrónicos, para computerizar libros de registro en oficinas o para publicar textos en un sitio web. La aplicación de OCR permite editar el texto, buscar una palabra o frase, almacenar de manera más compacta, mostrar o imprimir una copia limpia de marcas del escaneo y aplicar técnicas como traducción automática, deletreado o minería de textos.

OCR se fundamenta en principios de reconocimiento de patrones, inteligencia artificial y visión por computador. A continuación presentamos un posible esquema de algoritmo OCR (figura 9):

1. Binarización (de la imagen): transformar la paleta de colores a blanco y negro (puros).
2. Fragmentación o segmentación de la imagen: detectar los caracteres para tratarlos por separado.
3. Adelgazamiento de las componentes: perfilado de los contornos para conseguir una figura más regular.
4. Comparación con patrones: comparar los caracteres conseguidos con patrones de referencia (buscando un nivel máximo de coincidencia).

13.2. Python

Ha sido el lenguaje de programación utilizado para implementar el parserOCR y el parserMultilingüe.

En el momento de decidirnos por un lenguaje seguimos los criterios enumerados a continuación:

- Desarrollo ágil (lenguaje interpretado o de scripting): por tratarse de un programa de baja complejidad.

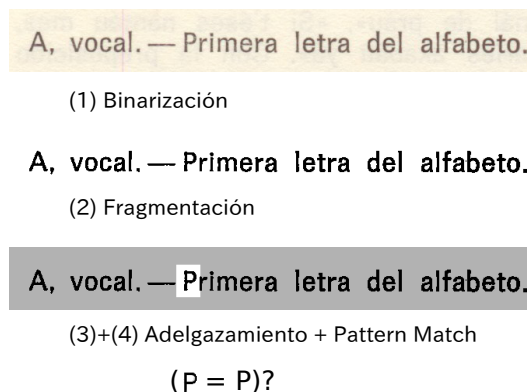


Figura 9: Algoritmo básico de OCR

- Limpieza y elegancia
 - Uso del intérprete interactivamente para probar/aprender
 - Filosofía Python: legibilidad y transparencia.
- Tratamiento de strings potente.
 - Uso de expresiones regulares.
 - Conectividad con el *SGBD* MySQL.

Python junto con Perl (diseñado originalmente para procesar texto: Practical Extraction and Report Language) han sido las dos alternativas que hemos valorado para programar el parser.

Ambos lenguajes de programación son interpretados y frecuentemente utilizados para crear scripts de automatización de tareas. La combinación de su potencial para el tratamiento de textos junto con el desarrollo ágil que proporciona los convierten en perfectos candidatos para esta tarea.

Finalmente, optamos por Python ya que no tenemos experiencia previa con ninguno de ellos pero nos convence más, a primera vista, la limpieza del código de Python.

13.3. Tecnología web y servidor de aplicaciones

En la presente sección presentaremos las tecnologías utilizadas en el desarrollo de la aplicación web así como una justificación de su elección.

13.3.1. Tecnología servidor. Java (JSP)

PHP es una de las tecnologías más extendidas para el diseño de páginas web dinámicas. Fue una alternativa que valoramos, pero acabamos descartando porque nuestra experiencia con PHP nos había dejado la sensación de que se



Figura 10: HTML+CSS+AJAX

trataba de un lenguaje poco claro a la hora de estructurar y separar la capa lógica de la capa de presentación.

Como Java ha sido el lenguaje utilizado durante la carrera, buscamos una de las soluciones Java para diseño web: JSP. Este lenguaje, considerado la solución Java del PHP, nos ofrecía las mismas ventajas que el desarrollo ágil de PHP pero permitiéndonos implementar una capa de dominio con el potencial de Java (y usando nuestra experiencia codificando en dicho lenguaje).

Además, podíamos acompañar Java del JDBC (Java Database Connectivity), API que permitía abstraernos de la capa de datos. Esto nos facilitaba la tarea de desvincular el proyecto de un SGBD concreto y nos ayudaba a obtener una gestión transparente de las conexiones a la base de datos.

13.3.2. Tecnología de la vista. HTML+CSS+AJAX

Como aplicación web, el resultado final que recibe el usuario es un fichero HTML. Para dar forma a las vistas aplicamos hojas de estilo CSS sobre dicho fichero.

Además, utilizamos tecnologías basadas en JavaScript para la carga dinámica de contenido en ciertas secciones de la vista. Concretamente, el framework JQuery como apoyo a la codificación de nuestras propias funciones. Dicha comunicación asíncrona entre la vista del cliente y el servidor está englobada bajo el concepto de AJAX¹⁵.

13.4. Sistema de Gestión de Bases de Datos (DBMS)

El SGBD es el encargado de gestionar la información almacenada en una base de datos y, por tanto, una pieza clave en el diseño de una aplicación.

En el entorno web, dentro de las alternativas de software libre, existe una disputada competencia entre MySQL y PostGreSQL. Ambos SGBD son las

¹⁵Asynchronous JavaScript And XML

opciones valoradas para este proyecto.

13.4.1. MySQL frente a PostGreSQL

MySQL se caracteriza por su velocidad, dirigido a aplicaciones con bases de datos de complejidad media y sin restricciones de integridad complejas en la BD.

PostGreSQL destaca un diseño que cumple con mayor fidelidad el estándar SQL, con un potente sistema de restricciones de integridad, soporte de transacciones, ACID y gran escalabilidad.

MySQL	PostGreSQL
Velocidad	Potencia y escalabilidad
Ampliamente usado	
Gran variedad de herramientas (phpmyadmin)	
GPLv2	BSD
InnoDB: ACID y Foreign Keys	Soporte nativo ACID

Cuadro 1: Comparativa MySQL frente a PostGreSQL[24, 25]

Dadas las características de nuestra aplicación, nos decantamos por la facilidad de gestión que nos aporta MySQL. Sus características funcionales son suficientes y tenemos experiencia utilizando este gestor. Además, su extendido uso nos facilitará el proceso de despliegue, por la multitud de servicios de hosting que han escogido MySQL como su gestor de bases de datos.

13.4.2. MySQL: motores de almacenamiento

La elección de un motor de almacenamiento diferente a MyISAM (por defecto en MySQL), nos la planteamos al descubrir que éste no proporcionaba ni restricciones foráneas (Foreign Keys) ni soporte para transacciones (ACID¹⁶). Ambas cosas podían facilitarnos mantener la integridad de nuestro contenido.

¹⁶ACID: en bases de datos se denomina ACID a un conjunto de características necesarias para que una serie de instrucciones puedan ser consideradas como una transacción.

- **Atomicidad:** es la propiedad que asegura que la operación se ha realizado o no, y por lo tanto ante un fallo del sistema no puede quedar a medias.
- **Consistencia:** es la propiedad que asegura que sólo se empieza aquello que se puede acabar. Por lo tanto se ejecutan aquellas operaciones que no van a romper la reglas y directrices de integridad de la base de datos.
- **Aislamiento:** es la propiedad que asegura que una operación no puede afectar a otras. Esto asegura que la realización de dos transacciones sobre la misma información sean independientes y no generen ningún tipo de error.
- **Durabilidad:** es la propiedad que asegura que una vez realizada la operación, ésta persistirá y no se podrá deshacer aunque falle el sistema.

En situaciones de actualización de la información las claves foráneas se actualizarán automáticamente en la capa de datos, evitando proporcionar información extra a la capa de dominio y, por ello, reduciendo el acoplamiento entre capas.

Por otra parte, cuando una actualización falle en una fase intermedia, el soporte de transacciones garantiza que no se aplicarán los cambios dejando la base de datos en su estado inicial.

El motor de almacenamiento proporcionado por MySQL que cumplía nuestros requisitos es el InnoDB, así que lo escogimos para el desarrollo del proyecto.

Nota: La única funcionalidad a la que renunciamos por descartar MyISAM fue la denominada búsqueda Full-Text (ver sección 16.2.4).

13.4.3. Bases de datos relacionales frente a Bases de datos XML

Se trata de una alternativa que contemplamos cuando valoramos la opción de almacenar las definiciones como XML.

Este método nos proporcionaba una gran flexibilidad a la hora de estructurar el contenido de ésta (introducir ejemplos de uso, palabras relacionadas, variantes locales, etc). Sin embargo, en nuestro caso particular del contenido del diccionario, nos obligaba a hacer un paso más en la estructuración de las definiciones. Con los errores de digitalización que había nos arriesgábamos a unos resultados que no justificaban todo este trabajo, así que acabamos descartando esta alternativa.

En cualquier caso creemos interesante hacer una mención, dado que nos ha resultado muy interesante conocer la existencia de bases de datos nativas XML. En la figura 11 se muestra un posible diseño para una base de datos con soporte nativo XML. Además acompañamos un ejemplo del contenido del fichero XML (algoritmo1) y la definición de su XMLSchema¹⁷(algoritmo 2).

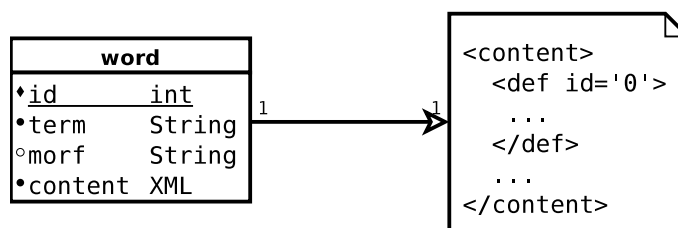


Figura 11: Diseño BD XML

Ventajas de BD XML:

- Al tratar con información semiestructurada (como es el caso de las definiciones), cuya estructura es lo suficientemente variable como para que almacenarla en una base de datos relacional suponga un elevado número

¹⁷XMLSchema es un lenguaje de esquema utilizado para describir la estructura y las restricciones de los contenidos de los documentos XML.

Algorithm 1 Ejemplo de fichero XML

```
<content>
  <definition id="1" class="required-one">
    Temperatura corporal o ambiental elevada,
    superior a la normal.
    <ar>calor</ar><ar>cal</ar>
    <ca>calor</ca>
    <es>calor</es>
    <fr>chaleur</fr>
  </multilang>
  <use-example>...</use-example>
</definition>
...
<definition id="N">
  Afecto, buena acogida.
</definition>
</content>
```

de columnas con valores nulos (gasto innecesario de espacio) o una gran cantidad de tablas (ineficiente).

- La obtención de la información puede ser más rápida cuando el acceso es secuencial. En el caso de las definiciones, al acceder por término o identificador bastaría con consultar una única tabla para obtener los resultados. Sin embargo, con el diseño de la base de datos relacional deben hacerse consultas a varias tablas uniendo los resultados.

Ventajas de BD relacional:

- Potencial para realizar búsquedas complejas (SELECTs).
- Rendimiento para accesos aleatorios. Por ejemplo para la búsqueda multilingüe.

Referencias:

1. Storing Data in a Native XML Database. <http://www.rpbouret.com/xml/XMLAndDatabases.htm#datainnative>
2. Bases de datos nativas XML - Wikipedia. http://es.wikipedia.org/wiki/Bases_de_datos_nativas_XML
3. Cover Pages: XML and Databases. <http://xml.coverpages.org/xmlAndDatabases.html>

Algorithm 2 XMLSchema del fichero XML de una palabra

```
<?xml version="1.0" encoding="UTF-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <!-- definition of attributes -->
  <xs:attribute name="id" type="xs:positiveInteger"/>

  <!-- definition of simple elements -->
  <xs:element name="ar" type="xs:string"/>
  <xs:element name="be" type="xs:string"/>
  <xs:element name="ca" type="xs:string"/>
  <xs:element name="es" type="xs:string"/>
  <xs:element name="fr" type="xs:string"/>
  <xs:element name="text" type="xs:string"/>
  <xs:element name="use-example" type="xs:string"/>

  <!-- definition of complex elements -->
  <xs:element name="definition">
    <xs:complexType>
      <xs:attribute ref="id" use="required"/>
      <xs:sequence>
        <xs:element ref="text" use="required"/>
        <xs:element ref="ar" minOccurs="0"/>
        <xs:element ref="be"/>
        <xs:element ref="ca" minOccurs="0"/>
        <xs:element ref="es" minOccurs="0"/>
        <xs:element ref="fr" minOccurs="0"/>
      </xs:sequence>
    </xs:complexType>
  </xs:element>

  <xs:element name="word">
    <xs:complexType>
      <xs:element ref="definition" maxOccurs="unbounded"/>
    </xs:complexType>
  </xs:element>

</xs:schema>
```

4. Otras bases de datos. <http://ict.udlap.mx/people/carlos/is341/bases12.html>
5. Almacenamiento de XML en BD relacionales. <http://modelobdpnfi.wordpress.com/2010/03/20/almacenes-de-datos-vs-bd-xml/>

14. Arquitectura de la aplicación web

La arquitectura de una aplicación define el esquema estructural fundamental de la organización del sistema de software, que consiste en subsistemas, sus responsabilidades e interrelaciones. La presente sección describe el diseño fijado para la aplicación, lo cual incluye: patrones de diseño utilizados, componentes del diseño interno, diagrama de navegación, diseño de la vista principal...

14.1. Patrones de diseño

“Cada patrón describe un problema que ocurre una y otra vez en nuestro entorno y, a su vez, describe el fundamento de la solución para dicho problema, de un modo que dicha solución puede utilizarse en infinidad de situaciones, sin hacer dos veces lo mismo.”

Christopher Alexander, arquitecto (1977)

14.1.1. MVC[9]

El Modelo-Vista-Controlador (MVC) es un patrón arquitectónico que aísla la capa de dominio de la interfaz de usuario, permitiendo un desarrollo desacoplado y un mantenimiento más sencillo. Su nombre viene dado por los tres componentes lógicos que conforman el patrón (ver figura 12). A continuación describimos su función:

- Modelo: gestiona la información y notifica a los observadores cuando la información cambia.
- Vista: procesa el modelo para transformarlo en un formato que permita la interacción con el usuario.
- Controlador: recibe la entrada de datos e inicia la respuesta invocando a los objetos del modelo.

14.1.2. FrontController

El FrontController se trata de un componente que centraliza la gestión de peticiones. Dentro de la aplicación es el elemento que recibe, procesa y decide a que controlador específico delegar la tarea. Concretamente, utiliza un parámetro “action” para decidir que operación ha solicitado el usuario.

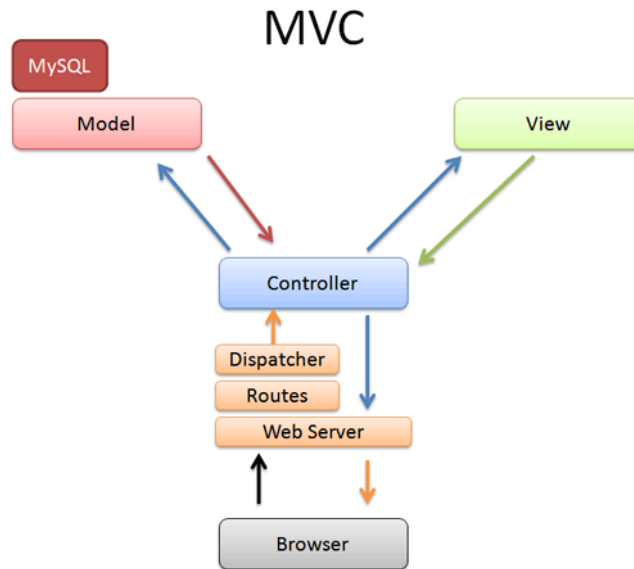


Figura 12: Patrón Modelo-Vista-Controlador

La implementación se realiza mediante una asociación uno a uno de códigos-acciones, por ejemplo, el código 3 corresponde a la acción de obtener una definición aleatoria.

Contexto: El sistema requiere un punto de entrada centralizado para la gestión de peticiones.

Problema: Sin dicho punto central, se duplica el código común a múltiples peticiones. Cuando el código de control está entremezclado con el código de creación de la vista, la aplicación pierde modularidad y cohesión.

- Evitar lógica de control duplicada.
- Aplicar lógica común a múltiples peticiones.
- Centralizar los puntos de acceso al sistema.
- Separar la lógica del sistema de la vista.

Solución: Utilizar FrontController como punto de entrada de las peticiones que realice una gestión inicial de éstas. El FrontController centraliza la lógica de control que de otro modo se duplicaría y gestiona el proceso de tratamiento de peticiones (validación de los datos de entrada, comprobación de permisos. . .).

14.1.3. DispatcherView

El DispatcherView añade un componente a la arquitectura del sistema que se encarga de generar nuevas vistas sin necesidad de invocar a la capa de dominio. De este modo, se delega al cliente la renderización de pequeños cambios, evitando nuevas peticiones al servidor.

En el sistema lo aplicamos para mostrar/ocultar información extra de la vista como la ayuda contextual o la licencia bajo la que está publicado el software.

Problema: Se quiere que una vista capture una petición y genere una respuesta que requiere ejecutar poca o nada lógica de negocio antes de generar la (nueva) vista.

Fuerzas:

- Se tienen vistas estáticas
- Las vistas se han generado desde un modelo existente de presentación
- Hay vistas independientes de la capa de dominio (o alguno de sus servicios)
- La capacidad de proceso es limitada

Solución: Aplicar DispatcherView con la vista como punto de acceso inicial de una petición. Si es necesario realizar alguna operación, de baja complejidad, es gestionada por las vistas.

14.1.4. InterceptingFilter

El InterceptingFilter es un componente que filtra determinadas peticiones realizadas al sistema para comprobar si se cumplen los requisitos de seguridad definidos.

En el sistema existen operaciones restringidas a usuarios autenticados como añadir, modificar y eliminar definiciones; por tanto, en cada ocasión que la aplicación reciba una petición de este tipo comprobará que se trata de un usuario válido y que posee los permisos suficientes para poder realizar la operación asociada.

Problema: Quieres interceptar y manipular una petición y/o una respuesta antes y después de que la petición sea procesada con el fin de determinar si:

- El cliente dispone de una sesión válida
- El curso de la petición viola alguna restricción
- La aplicación soporta el navegador del cliente
- El cliente ha utilizado una codificación especial para enviar los datos
- La petición está cifrada o comprimida

Fuerzas:

- Se busca una gestión centralizada y común entre peticiones
- Se quieren componentes que procesen a priori y a posteriori de manera desacoplada manteniendo un servicio central de gestión de peticiones
- Se desea que los componentes de pre y postprocesado sean independientes del resto y autocontenidos para facilitar su reutilización.

Solución: Usar un `InterceptingFilter` como filtro que pueda activarse/desactivarse para que pre y postprocese las peticiones y respuestas. Un `FilterManager` combina los filtros en cadena, delegando el control al filtro apropiado.

14.2. Diseño de la vista

La figura 13 muestra la estructuración de la vista principal de la aplicación web. Se compone de tres bloques principales:

1. header: contendrá el título, el menú de navegación y el selector de idioma.
2. content: mostrará el contenido generado por una búsqueda, una modificación, etc.
3. footer: alberga información sobre los derechos de copia y estadísticas de visitas.

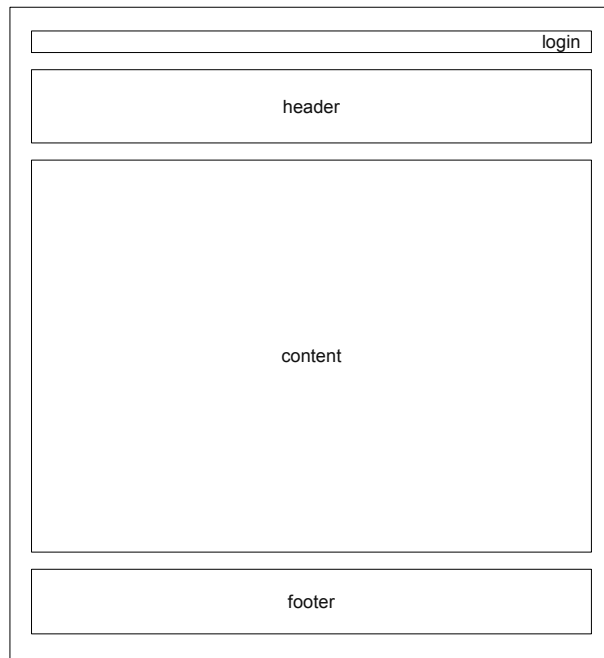


Figura 13: Diseño de la vista

Puede entenderse como un puzle de cuatro piezas: login, header, content y footer.

- El footer permanece invariable y siempre presente.
- El login se encarga de restringir el acceso a ciertos contenidos (modificar palabras).
- El header únicamente varía su menú de navegación (que incluye o no los botones de acceso a contenidos restringidos).
- El content se encaja con el resto de componentes mostrando la información generada.

14.3. Diagrama de componentes (diseño interno)

El diagrama UML con la extensión WAE¹⁸, adjunto en la figura 14, muestra los componentes que dan forma a la aplicación web. Se utilizan para modelar la vista estática de un sistema. Muestra la organización y las dependencias entre un conjunto de componentes.

¹⁸WAE: Web Application Extension (Extensión de Aplicaciones Web) es una extensión del UML que permite representar páginas web y otros elementos significativos en la arquitectura y el diseño interno de la capa de presentación de una web[33].

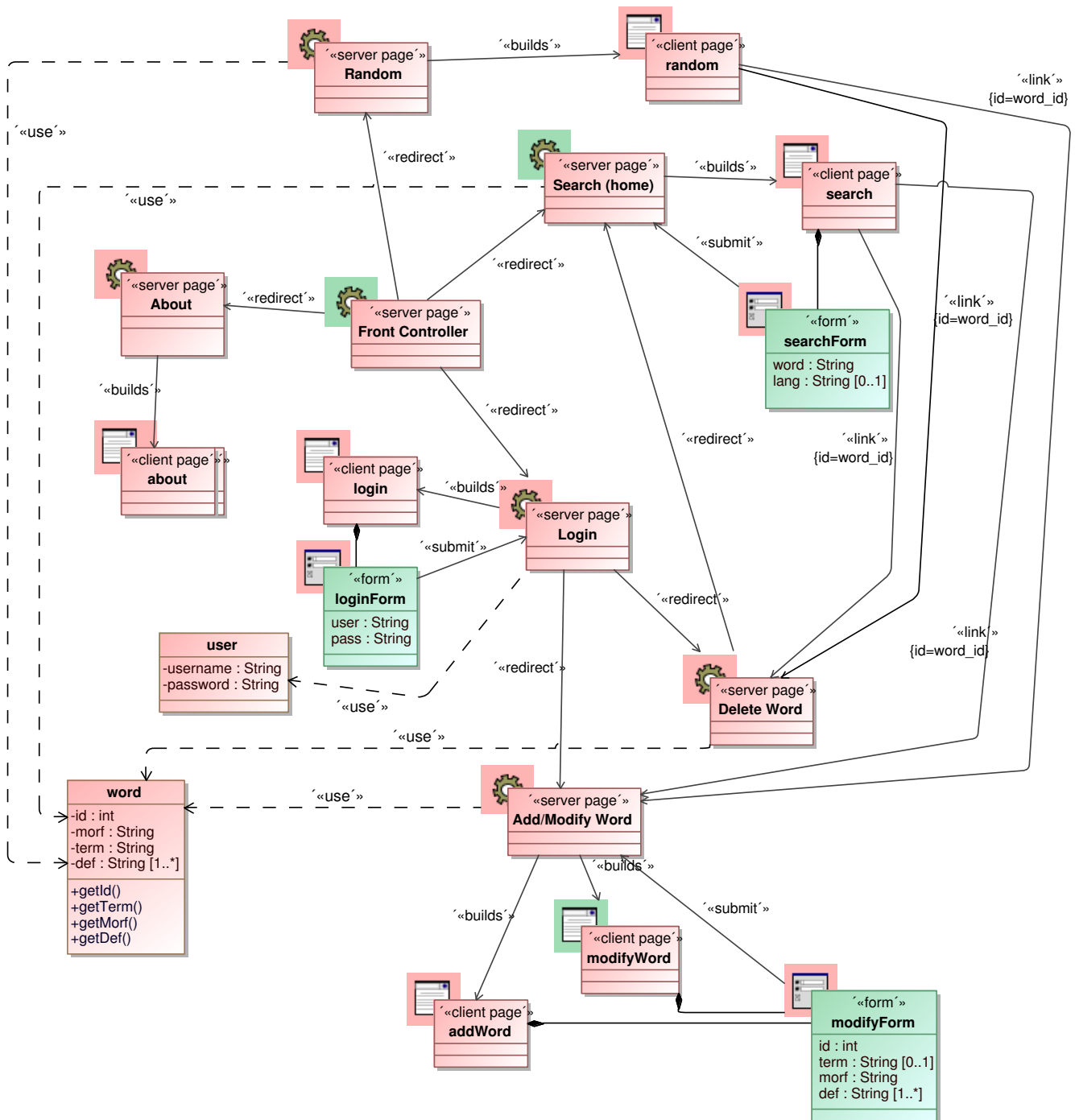


Figura 14: Diagrama de clases

14.4. Mapa de navegación

El mapa de navegación de la figura 15 muestra la distribución, estructuración e interrelación de los contenidos de la web.

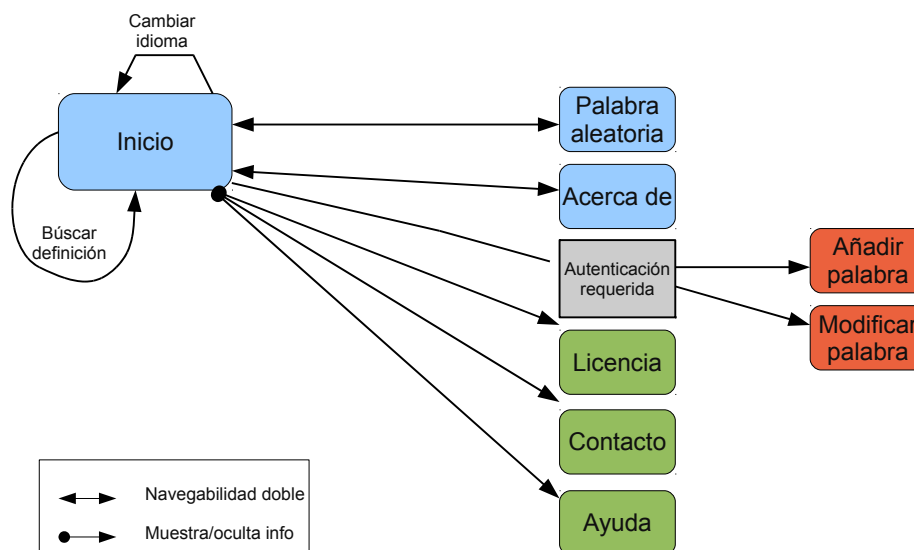


Figura 15: Mapa de navegación de la aplicación web

14.5. Diseño de la Base de datos[38]

El diseño de la base de datos ha evolucionado a la par que las funcionalidades de la aplicación. Puede verse su configuración final en la figura 16.

La funcionalidad de búsqueda multilingüe motiva la creación del conjunto de tablas multilang; sin embargo, debe aclararse que existe información duplicada entre la tabla multilang y el conjunto de multilang_[ar|be|ca|es|fr] fruto de la materialización de los atributos derivados. Se decidió materializar dicha información para optimizar la velocidad de consulta dado que el acceso es principalmente de lectura y no de modificación.

Descripción de las tablas

- Tablas principales
 - word: contiene la información relativa a la palabra.
 - word_definition: almacena la(s) definición(es) de la palabra (proporciona soporte para palabras polisémicas).
 - multilang: alberga la traducción del término a otros idiomas (aragonés, catalán, castellano y francés). Las variantes se agrupan separadas por comas (,).

- user: guarda la información de los usuarios.
- Tablas auxiliares: utilizadas durante la búsqueda estándar y multilingüe. Algunas palabras poseen varias posibles traducciones, de este modo podrán obtenerse resultados con cualquiera de las variantes (cada fila contiene una única variante del término o traducción).
 - multilang_ar: para el aragonés.
 - multilang_be: para el benasqués.
 - multilang_ca: para el catalán.
 - multilang_es: para el castellano.
 - multilang_fr: para el francés.

14.5.1. Restricciones de Integridad

En cuanto a la integridad referencial, tradicionalmente MySQL no soportaba el uso de claves foráneas (su motor predeterminado MyISAM), por eso decidimos usar el motor InnoDB. Permitirá mantener la relación entre el contenido de las tablas y evitar referencias huérfanas si en algún momento alguien realiza una modificación en los identificadores.

Por otra parte, hay un control mediante la restricción UNIQUE que evita que existan dos entradas con el mismo par <term, morf>; en situaciones así, el usuario debe añadir una nueva definición al par existente y no crear una nueva entrada en la tabla word.

Los campos word.validated y word.relation están diseñados con vistas a ampliaciones futuras. El primero de ellos permite a un administrador distinguir palabras revisadas de las que deben validarse. El segundo sirve para definir palabras relacionadas con esa definición, por ejemplo cuando esa palabra forma parte de una expresión hecha (*llevar* → *llevar adelante algo*).

14.5.2. Índices[12]

El uso de índices en una tabla es adecuado para situaciones en las que se realicen frecuentemente consultas (con resultados de pocos valores) en un conjunto de entradas de gran magnitud. Mediante la aplicación de índices se consigue una mejora importante en el rendimiento, dado que el número de accesos a disco disminuye drásticamente.

Sin un índice, MySQL debería comenzar desde la primera entrada y recorrer toda la tabla para encontrar las entradas relevantes. Cuanto mayor fuera el tamaño de la tabla, mayor el coste de búsqueda. Si la tabla posee un índice en las columnas utilizadas para la búsqueda, MySQL puede determinar rápidamente la posición en la cual obtener la información sin necesidad de recorrerla completamente. Para una tabla de 1000 filas esto supone, al menos, un acceso 100 veces más rápido que un acceso secuencial (como es el caso de las búsquedas en el diccionario).

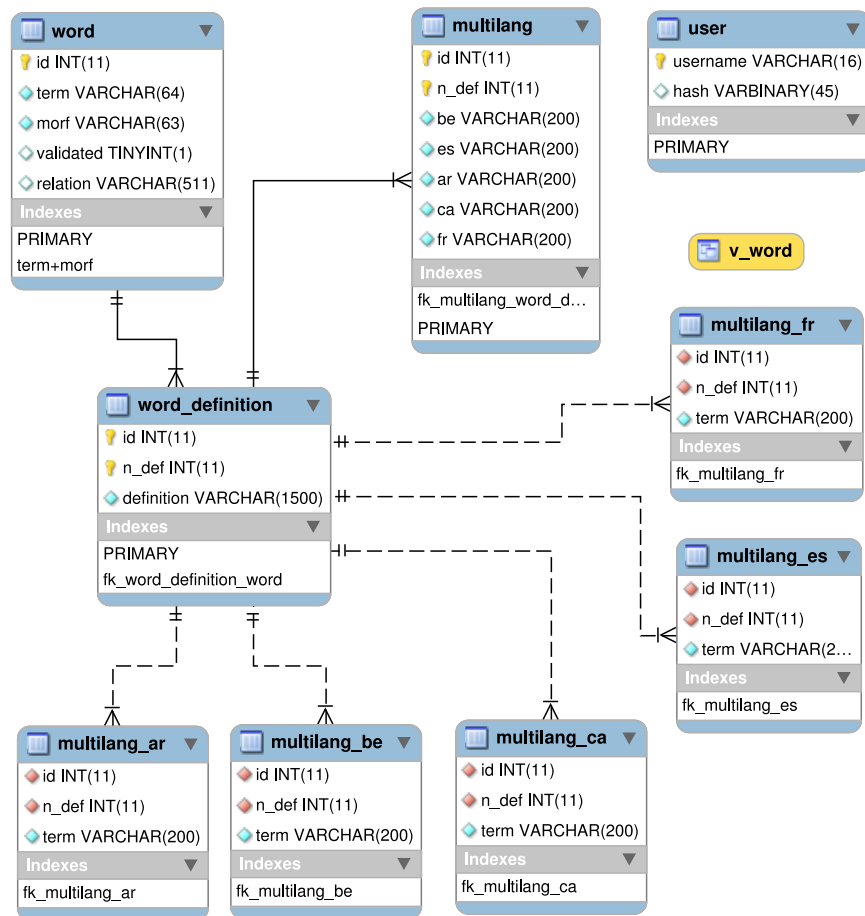


Figura 16: Diseño de la Base de Datos

La implementación de índices de MySQL que aplicamos en el sistema son los de tipo B-tree. Estos índices, cuyas entradas se almacenan en un árbol binario balanceado, pueden aplicarse en búsquedas con comparadores aritméticos ($<$, $>$, $=$, ...); tal es el caso de las consultas realizadas contra las tablas multilingüe durante las búsquedas de definiciones. Por tanto, como queda reflejado en el diseño de la base de datos (figura 16) nuestra decisión pasa por la creación de índices en dichas tablas.

Aunque su aplicación nos permite reducir los accesos a órdenes de magnitud logarítmicos, existen casos especiales en los que el uso de índices no está recomendado: situaciones en las que se necesita acceder a la mayoría de las entradas. En nuestro sistema, es el caso de la inicialización de las estructuras auxiliares de palabras cercanas lexicográficamente (sección 16.2.3), donde es más adecuado un acceso secuencial, por minimizar la cantidad de peticiones a disco.

Parte V

Implementación

15. Digitalización

15.1. Análisis del material

Partimos de un ejemplar en papel[29] que se compone de 524 páginas. De éstas, las 19 primeras son introductorias, así que el trabajo se centrará en las 505 restantes. Cada página se compone de dos columnas, en todas ellas está presente la numeración de página y en algunas de ellas aparece el título que determina un cambio de letra (“A”, “B”, “C”...); éstas dos últimas informaciones junto con alguna nota a pie de página esporádica deberán filtrarse por su irrelevancia como contenido.

Debe aclararse que dicho libro contiene **definiciones en castellano de palabras en benasqués**.

15.2. Alternativas

La valoración inicial arroja dos alternativas para alcanzar dicho objetivo:

- (a) Transcribir de manera manual el contenido (mecnografiar). [GRABADOR]
- (b) Escanear las 505 páginas, aplicar *OCR* y procesar la información (estructurarla) [OCR].

Aunque la primera opción evita una fase posterior en la que se estructura la información, se trata de una tarea repetitiva y que consumiría gran cantidad de tiempo sin aportar ningún aprendizaje, como puede verse en el cálculo estimado:

Opción grabador (tiempo estimado de realización):

$$diccionario_{chars} = 505páginas * \frac{800palabras}{página} * \frac{5carácteres}{palabra} = 1,767,500chars$$

$$t_{total.A} = 1,767,500chars / 200ppm * \frac{1hora}{60min} * \frac{1día}{8horas} \approx 21días$$

TOTAL = 21 días

Opción OCR (tiempo estimado de realización):

$$t_{scan} = 10lotes * (25s * \frac{50páginas}{lote} + \frac{60s}{carga.lote}) = 13,100s \approx 4horas$$

$$t_{OCR} = 505imágenes * \frac{75s}{imagen} = 37875s \approx 10horas$$

$$t_{parser} = t_{desarollo} + t_{ejecución} = 3días$$

$$t_{total.B} = t_{scan} + t_{OCR} + t_{parser} = 14horas * \frac{1día}{8horas} \approx 5días$$

TOTAL = 5 días

Queda claro que la segunda alternativa agiliza el proceso aunque requiera una fase de procesado (parsing) que transforme el texto plano en datos estructurados. Por lo tanto, optaremos por dicha alternativa que además, desde el punto de vista de un estudiante de ingeniería, es didácticamente más enriquecedora.

15.3. Digitalización mediante un software OCR

La sección actual describe el proceso realizado para extraer el texto de las imágenes escaneadas en la fase anterior.

Alternativas de software Dentro del amplio abanico de alternativas existentes nos centramos en analizar el siguiente conjunto con el fin de escoger la que finalmente aplicaremos a todo el contenido.

- ReadIris 9 Pro (incluido con el escáner)
- tesseract-ocr (HP Labs 1985-1995, actualmente por Google)
- free-online-ocr.com
- nlne.net

Para decidirnos por una de las alternativas hicimos una prueba con la primera página del diccionario que nos permitiese comparar las alternativas. La siguiente tabla muestra un resumen de los resultados:

	config. idioma	procesado lotes	online/desk	resultado
ReadIris 9 Pro	☑	☑	escritorio	bueno
tesseract-ocr	☒	☑	escritorio	muy malo
free-online-ocr.com	☒	☒	online	regular
n1ne.net	☑	☒	online	muy bueno

Cuadro 2: Comparativa software *OCR*

Las alternativas tesseract-ocr y free-online-ocr.com quedaron descartadas por su elevada cantidad de errores que convertían su salida en texto ininteligible.

El software de ReadIris superó satisfactoriamente la prueba, cometiendo errores únicamente en situaciones poco claras como "Í" (i mayúscula acentuada) frente "l" (L minúscula), y, además nos ofrecía la comodidad de ejecutarse en local y disponer de un modo guiado. Dicho modo permite supervisar el proceso, ayudando al programa a aprender en situaciones de duda, según la información proporcionada por el usuario.

A posteriori, con esta fase ya concluida, encontramos el servicio de n1ne.net, que incluimos a modo de referencia porque los resultados a la prueba fueron excelentes. Sin embargo, no recibimos respuesta a un envío real de un lote de unas 200 imágenes.

15.4. parserOCR / parserMultilang¹⁹

Consulte las secciones 11.2.3 y 11.2.4 para obtener detalles sobre la especificación de ambos scripts utilizados para extraer la información y estructurarla mediante una base de datos.

¹⁹Su código fuente se adjunta en el CD.

16. Aplicación Web

16.1. Aplicación de los patrones arquitectónicos

El uso de los patrones arquitectónicos descritos en la sección 14.1 se ha traducido en la implementación mostrada en la figura 17.

- La página 'index.jsp' realiza la función de FrontController, procesando las peticiones del usuario e invocando al controlador adecuado.
- Los controladores se eligen según el parámetro 'action' de la URI e invocan las operaciones del dominio.
- En el lado del cliente, se implementan mediante JavaScript los Dispatcher-View encargados de mostrar/ocultar información extra de la aplicación (ayuda contextual, licencia de uso...)
- El modelo se distribuye entre las clases Java: Entry, User, InOut, DBManager.

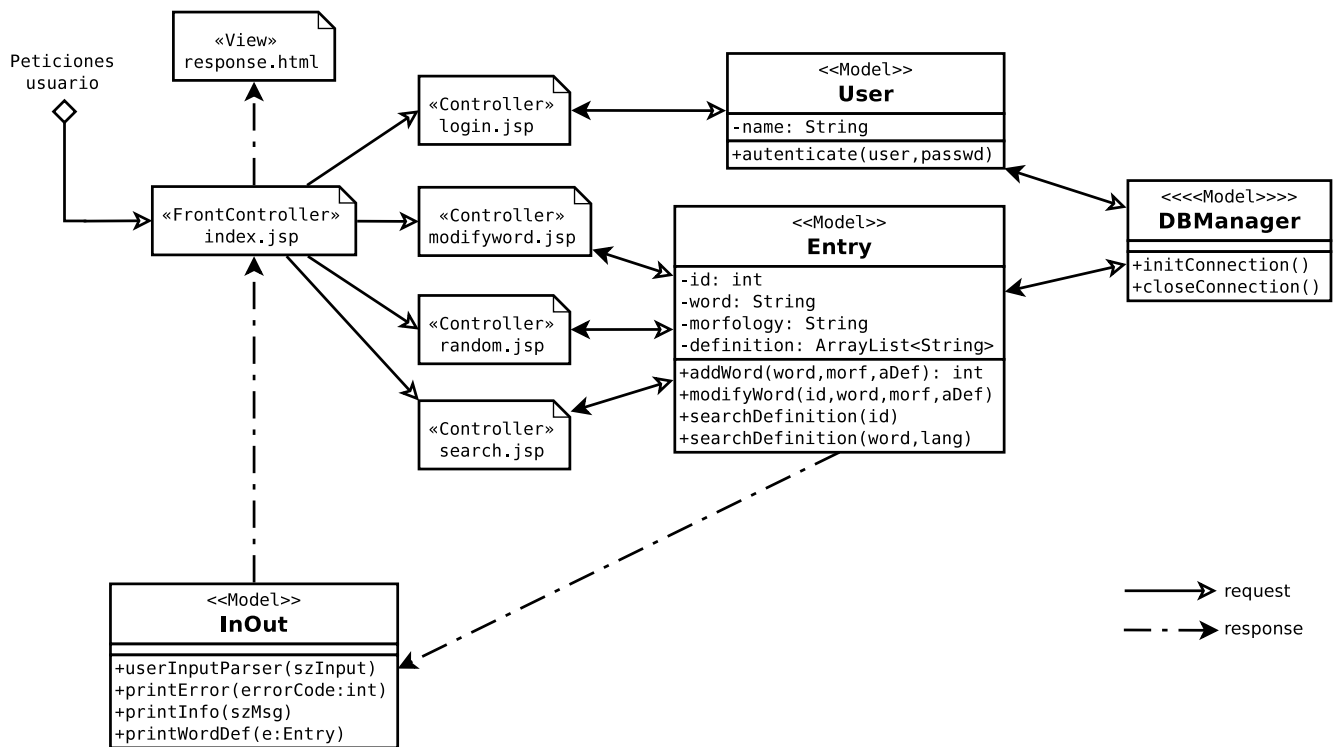


Figura 17: Esquema de la implementación

Algorithm 3 Primera aproximación palabras cercanas

```
1. buscar 15 anteriores:
    SELECT id,term FROM word WHERE UPPER(term) < UPPER($word)
    ORDER BY term DESC LIMIT 15
2. buscar 15 posteriores:
    SELECT id,term FROM word WHERE UPPER(term) > UPPER($word)
    ORDER BY term ASC LIMIT 15
3. unir respuestas
```

16.2. Implementación de las funcionalidades

16.2.1. Búsqueda estándar y multilingüe

Gracias al diseño de la base de datos se unificó la implementación de la búsqueda estándar y la multilingüe. Para ello se lanzan consultas SQL sobre las tablas multilang__[ar|be|ca|es|fr], escogiendo en función de la lengua de búsqueda (ver sección 14.5 para más información sobre las tablas).

Para dar soporte a las variantes de un término en benasqués y las diferentes traducciones a las otras lenguas pueden existir varias entradas para una misma palabra (identificador).

16.2.2. Palabra aleatoria

La justificación de esta funcionalidad viene motivada por uno de los objetivos del proyecto: la difusión del benasqués como lengua en peligro de extinción. Está inspirada en el artículo aleatorio que ofrece la Wikipedia y facilita que personas interesadas en ampliar su conocimiento del benasqués puedan curiosear y sorprenderse con el diccionario.

16.2.3. Palabras cercanas lexicográficamente

La implementación de la funcionalidad multilingüe supone un reajuste en el diseño de la operación de palabras cercanas lexicográficamente. Una primera aproximación aprovechaba el potencial de las bases de datos relacionales para ordenar los resultados obtenidos. Se basaba en el proceso descrito en el algoritmo3.

Sin embargo el coste de realizar estas consultas era elevado (dos ordenaciones) y poco flexible (hecho que se demostró al añadir la funcionalidad multilingüe). Por tanto decidimos cambiar el método volcando la responsabilidad sobre el servidor de aplicaciones. De este modo realizaríamos se reduce el acceso a disco sólo a la fase de inicialización y el resto de operaciones se realizarían contra una estructura almacenada en memoria.

La implementación descrita en el algoritmo4 hace uso de una tabla hash que contiene los pares <id, Entry> de las palabras existentes en la base de datos combinada con la implementación de la interfaz java.util.Comparator para la clase Entry.

Algorithm 4 Palabras cercanas Comparator

```
1 getNearWords(String szWord, String lang) {
2     [...]
3     ArrayList<Entry> aux = htWords.get(lang);
4     Entry e = new Entry(-1, szWord);
5     aux.add(e); // Add the entry looked up
6     Collections.sort(aux, new EntryComparator());
7     index = aux.indexOf(e);
8     aux.remove(index); // Delete the artificial word inserted
9     inf = Math.max(0, index - 15);
10    sup = Math.min(index + 15, aux.size());
11    for (index = inf; index < sup; index++)
12        l.add(aux.get(index));
13    return l;
14 }
```

16.2.4. Búsqueda contextual

La idea de crear esta funcionalidad surge directamente del feedback con los usuarios. Fue durante la presentación del proyecto en la Asociación Cultural Nogar²⁰, cuando un miembro del grupo SOFTWARE ARAGONES²¹ comentó que era de gran utilidad durante las traducciones poder contextualizar una palabra.

Además, nuestra experiencia en el aprendizaje de lenguas también reforzaba la importancia de darles contexto con el fin de integrarlas en nuestro vocabulario. Ilustraremos su uso mediante un ejemplo práctico:

Imaginemos que estamos traduciendo un texto y nos encontramos con la palabra “turbéres”. Si la buscamos obtendremos su definición: “Viento violento que lleva nieve polvina[...].” Ver esta palabra en múltiples contextos (en diferentes frases) puede ayudarnos a entender mejor su significado y, por tanto, a encontrar una traducción más fiel.

Para ello, basta con introducir la palabra “turbéres” y seleccionar la búsqueda contextual. Obtendremos las definiciones de palabras que contengan “turbéres” como, por ejemplo, “enreuláse” (perecer a causa de la nieve) con las frases «Bem pensá enreulámos de turbéres ke fébe.» (Temimos morir congelados de la ventisca que había).

Para su implementación se valoraron dos alternativas: Full-Text Search o consultas combinando la cláusula LIKE con expresiones regulares (REGEXP).

Full-Text Search[11, 20]

Se trata de una funcionalidad que analiza el contenido de las columnas que

²⁰La Asociación Cultural Nogar²⁰ es una asociación con sede en Zaragoza creada con el objetivo de defender la cultura y la lengua aragonesa. Más información en su web: <http://www.nogara-religada.org>

²¹SOFTWARE ARAGONES es un grupo de traducción de software al aragonés. Actualmente está trabajando en la traducción de la distribución GNU/Linux Ubuntu <http://groups.google.es/group/softaragones>

contienen texto (CHAR, VARCHAR o TEXT) indexando las palabras contenidas. Permite varios modos de búsqueda, pero nos centraremos en los dos más adecuados para esta situación: búsqueda booleana y búsqueda de lenguaje natural.

- La búsqueda booleana interpreta el string de búsqueda utilizando las reglas de un lenguaje de query especial. Dicho string, además de contener las palabras a buscar, puede especificar operadores para que una palabra aparezca obligatoriamente, o no pueda aparecer, etc. Además, posee la capacidad de ignorar palabras comunes *stopword* como conectores o preposiciones ('a', 'de', 'para'...).
- La búsqueda de lenguaje natural interpreta la frase literalmente, sin aplicar ningún operando. También aplica la lista de *stopwords*.

Conclusión

Full-Text Search es una herramienta que permite búsquedas complejas y versátiles que, sin embargo, son innecesarias para implementar esta funcionalidad (ver especificación en la sección 12.2.2). Además, en MySQL es una funcionalidad exclusiva del motor MyISAM, mientras que en la fase de diseño se optó por el motor InnoDB (cumple la especificación ACID y soporta claves foráneas).

El rendimiento es el punto a favor de Full-Text Search en este contexto pero dado el tamaño de la base de datos (inferior a 10.000 registros) asumiremos dicha penalización, decantándonos por el uso de LIKE, aunque anotaremos en la lista de trabajo futuro replantear esta decisión.

16.2.5. Login (security): Java Filters

Esta funcionalidad controla el acceso a las operaciones de modificación de las definiciones del diccionario, es decir, insertar, modificar y eliminar.

La implementación se basa en autenticación digest HTTP [15] sobre una conexión *SSL*, es decir, HTTPS. De este modo se cifran los credenciales del usuario evitando que se envíen como texto plano.

Por otra parte, la validación de los credenciales se realiza contra la base de datos que almacena un hash SHA-256²² de la contraseña del usuario [5]. Si se almacena en la base de datos el hash, en vez directamente la contraseña, un atacante no será capaz de recuperar la contraseña. El único modo de que supere esta barrera es un ataque de fuerza bruta, por ejemplo calculando el hash de todos las posibles contraseñas o con un diccionario que contenga los hash de contraseñas frecuentes.

Un último detalle es que se ha agrupado esta funcionalidad de login junto con las que permiten modificar palabras en un módulo. Este diseño por módulos permite activar o desactivar mediante un fichero de configuración. De esta manera, en un diccionario en el cual se quieran evitar modificaciones basta con que se desactive.

²²SHA-256: Secure Hash Algorithm (Algoritmo de Hash Seguro) es un sistema de funciones hash criptográficas.

16.3. Cuestiones generales de la implementación

16.3.1. Internacionalización: Java ResourceBundle[16, 7]

La tecnología Java dispone de una potente herramienta para crear aplicaciones multilingües: los *resource bundle*.

Los resource bundle contienen objetos locale²³ específicos (dependientes del idioma y/o país). Cuando la aplicación necesita un recurso locale, por ejemplo un texto, puede cargarlo desde un resource bundle apropiado para la locale del usuario. De este modo puede codificarse un programa, en gran parte, independiente de la locale del usuario aislando la información locale en resource bundles.

La aplicación de esta tecnología permite escribir aplicaciones que pueden:

- Ser traducidas fácilmente a diferentes idiomas.
- Gestionar múltiples locales simultáneamente.
- Añadir a posteriori soporte para nuevas locales.

En nuestra implementación concreta, la aplicación escoge por defecto la localización del navegador; sin embargo, un usuario puede seleccionar otro idioma mediante el menú desplegable.

16.3.2. Seguridad en aplicaciones web

Cross-Scripting (XSS), SQL Injection...

Toda aplicación web es susceptible de ser atacada por usuarios malintencionados aprovechando vulnerabilidades en el código: a través de formularios, modificando parámetros de la URI, capturando sesiones de usuario activas...

Nuestro objetivo fijado en este tema ha sido validar (y procesar) todo parámetro proporcionado por el usuario, en especial aquella información que a posteriori vaya a ser mostrada (búsquedas, resultados de añadir/modificar una palabra...):

- XSS: puede evitarse convirtiendo los símbolos que delimitan las etiquetas XML (< , >) en entidades HTML.
- SQL Injection[17]: puede prevenirse cumpliendo los siguientes principios:
 - Parametrizar todas las queries (usando el objeto PreparedStatement de Java)
 - Toda información dinámica (por ejemplo la enviada por el usuario) deben ser obligatoriamente utilizados mediante consultas parametrizadas.
 - La concatenación de cadenas de caracteres nunca debe utilizarse para crear SQL dinámico.

²³locale: préstamo del inglés que en informática se refiere al conjunto de parámetros que definen el idioma, país o región. Es utilizado en tareas de internacionalización (traducciones) y regionalización (cambios de formato en fechas, moneda, unidades de medida...)

Apache Tomcat - Security Manager

El SecurityManager de Java es el que permite a un navegador ejecutar un applet en su propio *sandbox*²⁴ con el objeto de prevenir que código no seguro acceda al sistema de ficheros local, se conecte a otro host que no sea desde el cual se ha cargado el applet, etc. Del mismo modo que el SecurityManager protege al usuario de la ejecución de applets no seguros en su navegador, el uso del SecurityManager para ejecutar Tomcat puede proteger al servidor de servlets troyanos, JSPs, JSP beans, bibliotecas tag o, incluso, de errores involuntarios.

16.3.3. Codificación (encoding, charset, cotejamiento...)

Cuando trabajas con lenguas distintas al inglés frecuentemente te enfrentas a problemas con la codificación de caracteres al intercambiar información entre el cliente, la aplicación web y la base de datos; este proyecto tampoco iba a librarse de ello.

Nuestra elección fue UTF-8. Destacar la importancia de especificar explícitamente la codificación utilizada en los JSPs, la cabecera del XHTML y la codificación aceptada en los formularios; de este modo pueden reducirse los problemas con la codificación durante los envíos y procesados de información.

A continuación detallamos hechos a tener en consideración cuando se escoge desarrollar con UTF-8 en las tecnologías que hemos escogido:

Python

Hasta la versión 3 (en el proyecto utilizamos la 2.6) Python trata de manera especial las cadenas de caracteres unicode (los cadenas estándar son ASCII de 7bits). Para trabajar en UTF-8 se deben realizar conversiones explícitas:

```
s = "hello_normal_string"
u = unicode( s, "utf-8" )
backToBytes = u.encode( "utf-8" )
```

Java

La codificación por defecto de la máquina virtual de Java es ISO-8859-1; y por tanto, debe indicarse explícitamente el cambio de codificación en:

- El conector Tomcat

```
<Connector port="8080" protocol="HTTP/1.1" [...]  
    URIEncoding="UTF-8" [...] />
```

- La configuración de los servlets (JSP)

```
// parameter-encoding = "UTF-8" defined in web.xml  
String charSet = application.getInitParameter("parameter-encoding");  
request.setCharacterEncoding(charSet);  
response.setCharacterEncoding(charSet);
```

²⁴sandbox: se refiere a un entorno aislado de ejecución que, por lo tanto, no puede afectar a la integridad del sistema.

- Los formularios

```
<form id="example" action="get.jsp" accept-charset="utf-8">  
    [...]  
</form>
```

MySQL

Para completar la configuración en UTF-8 seleccionamos una codificación para la base de datos. Existen codificaciones dependientes del idioma, por ejemplo `utf8_spanish_ci`, “ci” indica que son case insensitive²⁵.

En nuestro caso nos interesaba que fuera tanto independiente del idioma como case sensitive, por lo tanto elegimos `utf8_bin`. La razón es que en las definiciones pueden aparecer nombres propios y consideramos importante que la búsqueda los diferencie (Ballarín incluye múltiples entradas de toponimia del valle de Benasque).

²⁵case insensitive: no distinguen mayúsculas y minúsculas.
case sensitive: sí se distinguen mayúsculas y minúsculas.

Parte VI

Conclusiones

En esta sección presentaremos las conclusiones extraídas del proyecto, realizando una valoración global de éste.

Valoración general

El objetivo de potenciar la transmisión de una lengua es algo muy ambicioso y que debe plantearse como una meta a medio o largo plazo. En la vida del proyecto somos incapaces de tener perspectiva de la influencia de éste, pero los comentarios que hemos ido recibiendo de los usuarios nos han valido (y mucho) para valorar que existe una repercusión positiva en el medio.

El nivel de desarrollo de la aplicación cumple las necesidades funcionales para las que fue diseñada, aunque siempre queda la sensación de que continúan quedando infinidad de cosas por retocar, mejorar, añadir, cambiar... Hemos conseguido una aplicación robusta, de uso simple y amigable.

Refiriéndonos al aspecto puramente técnico, queremos destacar dos cuestiones muy presentes en el desarrollo del proyecto. La primera de ellas está relacionada con la tecnología empleada para la aplicación web (JSP y Tomcat); mientras que la segunda incumbe al tema de la codificación de caracteres.

Consideramos que el potencial de las aplicaciones web Java se ve ensombrecido por la dificultad de configuración de Tomcat. Durante el proyecto nos hemos encontrado multitud de lagunas en la documentación, comportamientos distintos según la plataforma, etc. Su curva de aprendizaje (ver figura 18) dificulta que dicha tecnología adquiera nuevos desarrolladores y, que el esfuerzo extra inicial, la descarte como tecnología apata para aplicaciones de baja complejidad.

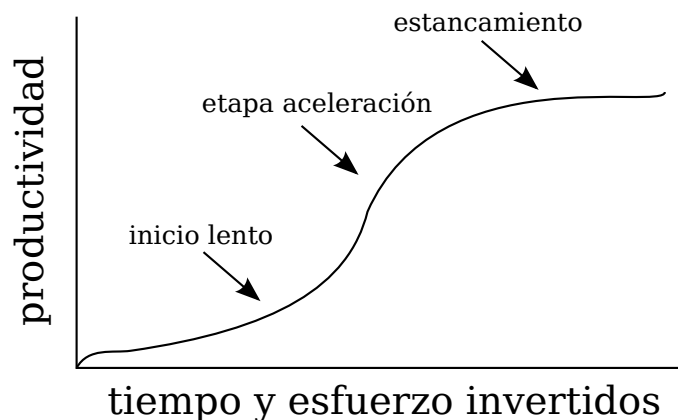


Figura 18: Curva de aprendizaje (JSP + Tomcat)[34]

En cuanto a la codificación de caracteres, creemos que continúa siendo una

tarea pendiente para la informática. A lo largo de toda la carrera, como desarrolladores, nos hemos encontrado en infinidad de situaciones en las que el uso de caracteres no ASCII ha acarreado problemas. Sigue sorprendiéndonos que algo que se usa de manera tan extendida como, por ejemplo, en la internacionalización del software no se haya resuelto de una manera más uniforme y unificada. Como se ha visto en la sección 16.3.3, cada tecnología requería un tratamiento específico de este aspecto.

Valoración personal

Estamos muy satisfechos por el trabajo realizado que nos ha permitido conocer con una mayor profundidad nuestra lengua, el benasqués; y más aún de haberlo hecho a través de la genial obra de Ángel Ballarín. Su trabajo no tiene precio, cuantas más horas invertíamos trabajando con su obra más conscientes éramos de la encomiable labor que realizó y que supo transcribir tan acertadamente.

Tras nuestra primera versión nos sabía a poco el aprovechamiento de la información recogida por Ballarín, y ese fue el desencadenante clave para continuar con la versión multilingüe. Este cambio nos ha permitido reflejar el estudio comparativo entre el benasqués y sus lenguas vecinas.

Por otra parte, en cuanto al aspecto técnico, llevar a cabo un proyecto completo de software nos ha ayudado a asimilar el proceso teórico descrito en la carrera. Y, además, ha cambiado nuestra visión sobre el desarrollo de software: su complejidad, la importancia de un buen análisis para prevenir dificultades en fases posteriores, un diseño coherente para facilitar las modificaciones posteriores... En definitiva, un buen desarrollo de software implica mucho más que saber programar.

La realización de este proyecto nos ha permitido conjugar las habilidades adquiridas a través de las diferentes materias de nuestra ingeniería. Conocimientos adquiridos sobre lenguajes regulares en la rama de la teoría de la computación; aptitudes y metodologías para el desarrollo de software aprendidas en PESBD; patrones de diseño presentados en ES2; referencias en el diseño de entornos web gracias a DSBW; la visión teórico-práctica sobre el funcionamiento interno de los parsers en Compiladores; y un largo etcétera que completa cada una de las materias que han integrado nuestra formación.

16.4. Trabajo futuro

Esta sección describe un conjunto de mejoras que deben considerarse para sistemas en entornos de producción. Además, se incluyen varias ampliaciones que por las características del proyecto quedaron fuera de su alcance, pero pueden ayudar a la mejora del conjunto de la aplicación.

Ampliaciones

16.4.1. WebService

Crear un método para el acceso al contenido desde otras aplicaciones (base para el desarrollo aplicaciones didácticas).

16.4.2. Sistema de notificación de errores

Permitir a usuarios no registrados notificar errores en el contenido de la definición.

16.4.3. Diccionario ilustrado

Adjuntar fotografías, imágenes u otros métodos gráficos que ilustren el contenido del diccionario. En concreto, la idea está vinculada a entradas del diccionario de herramientas o quehaceres tradicionales que han desaparecido de la vida cotidiana. Sería una modesta contribución a la memoria histórica.

16.4.4. Conjugador de verbos

Frecuentemente, los diccionarios permiten generar la conjugación de los verbos. Dicha utilidad permite comprobar su correcto uso en la diversidad de tiempos verbales.

Mejoras

16.4.5. Internacionalización del JavaScript

Para traducir los mensajes de error y de ayuda contextual. Puede utilizarse JQuery i18N²⁶.

16.4.6. Desarrollar un instalador

Que automatice el despliegue de la aplicación: configurando el servidor Tomcat, creando la base de datos y resolviendo las dependencias.

16.4.7. Usuarios y seguridad login

El sistema actual no incluye el módulo de gestión de usuarios de la aplicación, queda como trabajo futuro implementar esta funcionalidad.

En este área, existe una posible mejora para evitar ataques con diccionario contra el sistema de autenticación: realizar un hash de la contraseña con salt²⁷. Este sistema incluye un texto aleatorio (salt) generado para cada usuario de modo que en vez de almacenar el hash de su contraseña se aplica el hash a la combinación de ésta con el salt.

²⁶JavaScript Internationalization: <http://codingwithcoffee.com/?p=272>

²⁷Salt: http://www.owasp.org/index.php/Hashing_Java#Complete_Java_Sample

De esta manera, aunque dos usuarios tuviesen la misma contraseña en la base de datos se almacenarían claves distintas; además, se evitarían ataques con diccionarios de hash precalculados.

16.4.8. SGBD XML nativo + XSLT en la gestión de contenido

Sustituir el diseño actual de la base de datos por uno que incluya ficheros XML. De este modo podría darse una mayor versatilidad al formato de las definiciones incluyendo “inline” la traducción multilingüe, ejemplos de uso pero también pudiendo buscar según esos parámetros (ver sección 13.4.3).

Nomenclatura

- DBMS** DataBase Management System; se trata de sistemas software específicos dedicados a servir de interfaz entre la base de datos, el usuario y las aplicaciones que la utilizan.
- dpi** Los puntos por pulgada (ppp) del inglés dots per inch (DPI) es una unidad de medida para resoluciones de impresión, concretamente, el número de puntos individuales de tinta que una impresora o tóner puede producir en un espacio lineal de una pulgada.
- OCR** Reconocimiento Óptico de Carácteres, por sus siglas en inglés , Optical Character Recognition. Se trata de una tecnología que transforma imágenes de textos en los caracteres que los componen.
- PDF** Portable Document Format - Adobe®
- PNG** (Portable Network Graphics) es un formato gráfico basado en un algoritmo de compresión sin pérdida para bitmaps no sujeto a patentes.
- SGBD** Sistema de Gestión de Bases de Datos (ver DBMS)
- SSL** Secure Sockets Layer (Protocolo de Conexión Segura), es un protocolo criptográfico que proporciona comunicaciones seguras por una red, comúnmente Internet.

Referencias

- [1] *Atlas UNESCO de las Lenguas en Peligro en el Mundo*. <http://www.unesco.org/culture/es/endangeredlanguages/atlas>
- [2] *Dialectos del aragonés* - Wikipedia. http://es.wikipedia.org/wiki/Dialectos_del_aragon%C3%A9s
- [3] *Diccionari de la Llengua Catalana Multilingüe*. <http://www.multilingue.cat>
- [4] *Enciclopèdia Catalana, SAU*. <http://www.enciclopedia.cat>
- [5] *Hashing Java* - OWASP. http://www.owasp.org/index.php/Hashing_Java
- [6] *Homonimia y polisemia* - Wikilengua. http://www.wikilengua.org/index.php/Homonimia_y_polisemia#Distinci.C3.B3n_entre_homonimia_y_polisemia
- [7] *Internacionalización en Java: ResourceBundle*. <http://www.chuidiang.com/java/ResourceBundle/internacionalizacion.php>
- [8] *Java Platform SE 6 - API*. <http://java.sun.com/javase/6/docs/api/>
- [9] *Model-view-controller in Java platform* - Wikipedia. http://en.wikipedia.org/wiki/Model-view-controller#Java:_Java_Platform.2C_Enterprise_Edition_.28Java_EE.29
- [10] *MVC or MVP Pattern*. http://blogs.infragistics.com/blogs/todd_snyder/archive/2007/10/17/mvc-or-mvp-pattern-whats-the-difference.aspx
- [11] *MySQL 5.1 :: Full-Text Search Functions*. <http://dev.mysql.com/doc/refman/5.1/en/fulltext-search.html>
- [12] *MySQL Reference :: How MySQL Uses Indexes*. <http://dev.mysql.com/doc/refman/5.0/en/mysql-indexes.html>
- [13] *Publicaciones del Grupo de Estructuras de Datos y Lingüística Computacional*. http://protos.dis.ulpgc.es/art_gied.htm
- [14] *Real Academia Española*. <http://www.rae.es>
- [15] *RFC 2617 - HTTP Authentication*. <http://www.ietf.org/rfc/rfc2617.txt>
- [16] *Server-side Java: Internationalize JSP-based Websites*. <http://www.javaworld.com/javaworld/jw-03-2000/jw-03-ssj-jsp.html>
- [17] *SQL Injection Prevention*. http://www.owasp.org/index.php/SQL_Injection_Prevention_Cheat_Sheet

- [18] *Three Tier Arch vs MVC*. http://en.wikipedia.org/wiki/Three_tier_%28computing%29#Comparison_with_the_MVC_architecture
- [19] *Understanding Models, Views and Controllers / BetterExplained*. <http://betterexplained.com/articles/intermediate-rails-understanding-models-views-and-controllers/>
- [20] *Using MySQL Full-text Searching*. <http://devzone.zend.com/article/1304>
- [21] *Wikipedia*. <http://www.wikipedia.org>
- [22] *Wikipedia - OCR*. http://en.wikipedia.org/wiki/Optical_character_recognition
- [23] *WordReference.com*. <http://www.wordreference.com>
- [24] *PostgreSQL vs. MySQL*. http://danielpecos.com/docs/mysql_postgres/index.html. Version: 05 2010
- [25] *WikiVs - MySQL vs PostgreSQL*. http://www.wikivs.com/wiki/MySQL_vs_PostgreSQL. Version: 02 2010
- [26] AHO, Alfred V. ; LAM, Monica S. ; SETHI, Ravi ; ULLMAN, Jeffrey D.: *Compilers: Principles, Techniques, and Tools*. Addison-Wesley, 2007
- [27] ALUR, Deepak: *Core J2EE Patterns: Best Practices and Design Strategies*. 2. Prentice Hall, 2003
- [28] ANDOLZ CANELA, Rafael: *Diccionario aragonés. Aragonés-castellano y castellano-aragonés*. 4^ª. Librería General, 1977
- [29] BALLARÍN CORNEL, Ángel: *Diccionario del benasqués*. 2. Zaragoza : La Editorial, 1978. – ISBN 84-85-077-08-3
- [30] CASANELLAS BERGES, Fèlix: *En tu idioma, el caso colombiano*, FIB (UPC), Diplomarbeit, 2009. <http://hdl.handle.net/2099.1/7649>
- [31] CASES, R. ; MÀRQUEZ, L.: *Llenguatges, gramàtiques i autòmats. Curs bàsic*. 3. Edicions UPC, 2000
- [32] CASTAN, M. ; PARICIO, S.: Una lengua y una literatura invisibles: el caso del aragonés. In: *Letras aragonesas* (2008), Nr. 6. <http://docs.google.com/fileview?id=0B-j59vNz3mLfMmY3ZGQ2ZjEtMzQxYS00ZmRmLWEzZTgtMDc3NWVkMmIyOTYy&hl=en>
- [33] CONALLEN, J.: *Building Web Applications with UML*. 2. Addison-Wesley, 2003
- [34] DEWEY, Russ: *The Learning Curve / Cognition*. http://www.psywww.com/intropsych/ch07_cognition/learning_curve.html

- [35] FERRAZ Y CASTÁN, Vicente: *Vocabulario del dialecto que se habla en la Alta Ribagorza*. Madrid : Tipografía de la Revista de Archivos, 1934
http://catalleg.ub.edu/record=b1272246~S1*cat
- [36] FIGUEROLA, C. ; ALONSO, J.L. ; ZAZO, A. F.: Diseño de un motor de recuperación de la información para uso experimental y educativo. In: *BiD* (2000), Juni, Nr. 4. <http://www.ub.edu/bid/04figue2.htm>. – ISSN 1575–5886
- [37] FREIXES PÉREZ, Raquel: *Estudio i ampliación del proyecto educativo Aprendiendo Kiche*, FIB (UPC), Diplomarbeit, 2008. <http://www.aprendiendokiche.org>
- [38] GARCIA-MOLINA, Hector ; ULLMAN, Jeffrey D. ; WIDOM, Jennifer: *Database systems: the complete book*. Prentice Hall, 2009
- [39] LARMAN, C.: *Applying UML and Patterns*. 3. Prentice Hall, 2005
- [40] QUINTANA, Artur ; ARNAL, María L. ; TOMÁS, Xavier ; VÁZQUEZ, Juan J. u. a.: *Decálogo de conclusiones sobre la situación de la lengua aragonesa*. http://www.espaciopirineos.com/?option=com_content&view=article&id=457catid=3&Itemid=207. Version: April 2010
- [41] SANTANA, O. ; HERNÁNDEZ, Z. ; PÉREZ, J. ; RODRÍGUEZ, G. ; CARRERAS, F.: Diccionarios en soportes informáticos. In: *Cuadernos Cervantes de la Lengua Española* (1996), Dezember, Nr. 11. http://www.gedlc.ulpgc.es/art_ps/art25.pdf
- [42] SAURA RAMI, José A.: Ley de lenguas y política lingüística en Aragón. In: *Guayente* (2010), April, Nr. 86

ANEXO I - Análisis Diccionarios Digitales

Existe un amplio abanico de diccionarios que representan a diversidad de lenguas con un mayor o menor número de hablantes. Sin embargo, por cercanía lingüística, han sido el diccionario publicado por la RAE y el de la Enciclopèdia SAU los que hemos tomado como referencia.

Ninguno de los portales citados dispone de información sobre la implementación de sus servicios y tampoco respondieron a nuestros intentos de contacto para obtener detalles de las especificaciones del sistema o, incluso, la liberación de su código fuente.

Análisis de sus características funcionales

Ambos ofrecen un servicio de búsqueda simple, en el cual se introduce el término del cual se desea obtener su definición. Observamos que en la respuesta, para una misma entrada (p.ej. “a”) pueden existir varias palabras y cada una de ellas contener múltiples definiciones.

Por lo tanto:

- Una palabra como entidad además de por su grafía se identifica por su morfología (palabras homónimas).
 - Dentro de esta categoría existen las homógrafas y las homófonas. Mientras que el primer caso deberemos considerarlo durante la fase de diseño, el segundo es irrelevante porque al tener grafías diferentes no existe posible confusión en su almacenamiento [FUTURO: palabras homófonas asociadas]
- Una misma palabra puede poseer varias definiciones (palabras polisémicas).

Nota: La diferencia fundamental entre la polisemia y la homonimia está en el origen de las palabras, es decir, en su etimología.[6]

Diccionario de la RAE[14]

La búsqueda “por aproximación” y la “exacta” no presentan diferencias cuando se encuentra un término coincidente con la entrada del usuario. Sin embargo, muestran diferencias de comportamiento en caso contrario:

Por aproximación Intenta buscar palabras cercanas sustituyendo b/v, y/ll, cambiando tildes, etc. (generalmente cuestiones asociadas a errores ortográficos o tipográficos).

La búsqueda de “abión” nos devuelve “avión” y “habón”.

Exacta Simplemente muestra un aviso de que la palabra introducida es inexistente en su diccionario.

Sin signos diacríticos Muestra la pareja de palabras que se distinguen mediante un acento diacrítico²⁸.

Semejanza fonético-ortográfica La búsqueda de “vaca” arroja los siguientes resultados:

* baca * vaca * vacua

Diccionario de la Enciclopedia Catalana, SAU [4]

Explícitamente no dispone de una configuración manual para que el usuario modifique los criterios de búsqueda. De un modo automatizado muestra un resultado u otro en función de si obtiene términos coincidentes:

Exacta En caso afirmativo.

Listado de palabras cercanas En caso de no dar con el término introducido muestra una lista de palabras alfabéticamente cercanas.

Diccionario multilingüe de la Enciclopedia Catalana, SAU [3]

Incluye botones diferentes según la lengua deseada para realizar la búsqueda. Sigue el mismo comportamiento que el diccionario de catalán (listando las palabras cercanas en caso de no existir el término en la lengua elegida).

WordReference [23]

Se trata de una herramienta orientada a la traducción entre idiomas aunque también incluye diccionarios monolingües. Permite escoger el diccionario a través de un formulario con “radio-buttons”. Según si la búsqueda introducida arroja o no resultados muestra:

Exacta En caso de encontrar.

Lista de palabras similares Si no encuentra ninguna traducción presenta una lista de sugerencias de palabras a las que podías referirte (por similitud lexicográfica).

²⁸Dicho de un signo ortográfico: Que sirve para dar a una letra o a una palabra algún valor distintivo. *El adverbio más lleva acento diacrítico frente a la conjunción mas.*

ANEXO II - Manual de usuario

Uso del diccionario digital

Esta sección está dirigida a los usuarios finales de la aplicación. Mientras utiliza la aplicación puede encontrar una información similar pulsando sobre el enlace de ayuda.



Figura 19: Vista general de la aplicación web

Búsqueda estándar

Mediante el campo de texto introduzca el término cuya definición desee encontrar. Si la búsqueda es coincidente con alguna(s) palabra(s) de nuestra base de datos se mostrará la entrada. En caso contrario aparecerá una lista de las palabras más cercanas lexicográficamente. En dicha lista se le permitirá seleccionar cualquiera de los términos para obtener su definición asociada.

Búsqueda a partir de una palabra en otro idioma

Por otra parte, si desea obtener la traducción de un vocablo al benasqués puede introducir dicha palabra y hacer uso de los botones disponibles a la derecha del campo de búsqueda. Estos aparecen etiquetados con "ar", "ca", "es" y "fr" para referirse al aragonés, catalán, español y al francés, respectivamente (figura 20).

Obtener una definición al azar

Además tiene la posibilidad de utilizar la búsqueda aleatoria, es decir, automáticamente se le mostrará una entrada al azar; de este modo podrá conocer nuevos vocablos (figura 20).

Ignorar acentuación

Activando esta casilla la búsqueda ignorará los acentos al generar el resultado. Si por ejemplo introduce 'mes' le devolverá los resultados 'mes' (del año) y

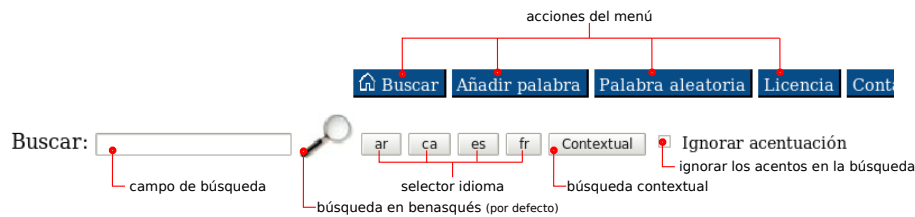


Figura 20: Detalle formulario de búsqueda y menú

'més' (adverbio de cantidad) (figura 20).

Modificación del contenido²⁹

Una vez registrado en la aplicación (mediante el formulario de login), dispondrá de acceso a las funcionalidades de modificación de contenido: añadir, modificar y eliminar palabras. La función de añadir aparecerá en el menú de navegación mientras que las de modificar y eliminar junto al término de la palabra (tras el resultado de una búsqueda).

Añadir/modificar una palabra

1. (Añadir) Pulse sobre la opción “Añadir palabra” en el menú de navegación. (Modificar) Una vez buscada la palabra que desea modificarse, pulse sobre la opción “modificar” que aparece como superíndice de la palabra (figura 21).

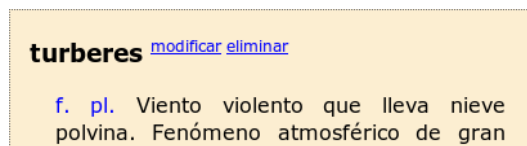


Figura 21: Opciones modificar/eliminar palabra

2. Se le mostrará un formulario el cual debe completar con la nueva información de la palabra (figura 22).

²⁹ *NOTA: Para acceder a esta funcionalidad debes disponer de una cuenta de usuario. Si es usted el administrador del sistema consulte la sección 16.4.8.*

Modificar palabra:

Término*	<input type="text" value="turberes"/>
Morfología	<input type="text" value="f. pl."/>
Definición*	<div><div>Viento violento que lleva nieve polvina. Fenómeno atmosférico de gran importancia, por lo dañino para personas y animales, y porque rellena las canales de</div><div>[+]</div></div>
<input type="button" value="Modificar palabra"/>	

* Indica campo obligatorio.

Figura 22: Formulario añadir/modificar palabra

- Una vez completado pulse el botón de enviar formulario.
Si hay algún problema con la información completada el sistema le advertirá y deberá realizar las modificaciones necesarias.
- La nueva palabra o las modificaciones sobre ésta ya se han completado (el sistema le mostrará un mensaje confirmándolo).

Eliminar una palabra

- Una vez buscada la palabra que desea eliminarse, pulse sobre la opción “eliminar” que aparece como superíndice de la palabra (figura 21).
- Debe confirmar la acción para que el sistema la lleve acabo.

Administración del diccionario digital

La presente sección está dirigida a los administradores de la aplicación.

Instalación

Los puntos 1 y 2 se refieren a las dependencias que deben cumplirse para que la aplicación funcione correctamente.

- Bibliotecas Tomcat:
Opción A) Instalar paquete `libmysql-java` (distribuciones basadas en Debian)
Opción B) Copiar:

- `naming-factory-dbc.jar` en
`/usr/share/tomcat5.5/common/lib/` (Tomcat 5.5)
`/usr/share/tomcat6/lib/` (Tomcat 6)
 - `mysql-connector-java.jar` en
`/usr/share/tomcat5.5/common/lib/` (Tomcat 5.5)
`/usr/share/tomcat6/lib/` (Tomcat 6)
2. Software: (distribuciones basadas en Debian: `aptitude install sun-java6-jdk apache2 tomcat6 phpmyadmin mysql-server`)
 - a) Servidor Tomcat (≥ 5.5)
 - b) SDK Java (≥ 6)
 - c) MySQL
 - d) Apache (opcional)
 - e) phpMyAdmin (opcional)
 3. (Crear) Importar BD: `_var/DB_creation_v2.sql`
 4. Configurar la cuenta de usuario que accederá a la base de datos:
 - a) Crear el usuario en MySQL y añadir los permisos para la base de datos 'dict' {SELECT; INSERT; UPDATE; DELETE}
 - b) Actualizar la información en `META-INF/context.xml`
 5. Configurar la política de seguridad del Tomcat (`/etc/tomcat6/policy.d/04webapps.policy`):


```
grant {
    permission java.lang.RuntimePermission
        "accessClassInPackage.org.apache.tomcat.dbcp.*";
};
grant codeBase "file:/var/lib/tomcat6/webapps/jspWebDict/" {
    permission java.net.SocketPermission "localhost:3306", "connect,resolve";
};
```
 6. Configurar la codificación de Tomcat en el fichero `server.xml` (y el conector `mod_jk` si se utilizase):


```
<Connector port="8080" protocol="HTTP/1.1" [...]
    URIEncoding="UTF-8" [...] />
```
 7. Configurar Tomcat para que acepte conexiones seguras (HTTP over SSL).
 Con el fin de conseguir login seguro.
<http://tomcat.apache.org/tomcat-5.5-doc/ssl-howto.html>
<http://tomcat.apache.org/tomcat-6.0-doc/ssl-howto.html>
 [Los puertos por defecto son 8080 y 8443. Si se desean utilizar otros puertos cambiar fichero de configuración...TODO!]

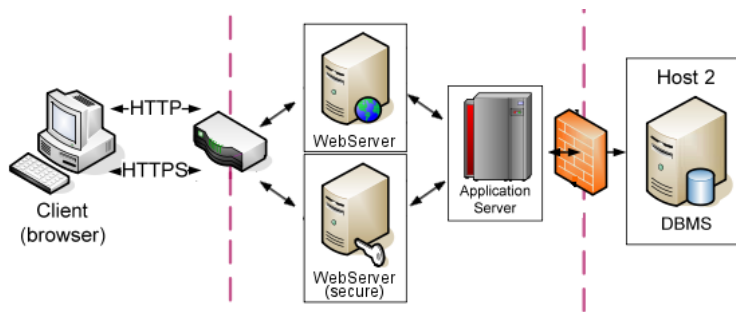


Figura 23: Diagrama de despliegue

8. Desplegar la aplicación:

Opción A) Copiar `jspWebDict.war` en el directorio `appBase` (por defecto `/var/lib/tomcat6/webapps`)

Opción B) Mediante el manager `http://localhost:8080/manager/html/` (Para utilizarlo debe crearse un usuario con rol 'manager' en el fichero `/etc/tomcat6/tomcat-users.xml`)

Cuestiones sobre la configuración del despliegue

Por cuestiones de rendimiento es frecuente combinar el servidor de aplicaciones (Tomcat) con el servidor web (Apache). Para esta aplicación se ha probado dicha combinación mediante el módulo “`mod_jk`”. Al aplicar este modelo, el servidor de aplicaciones queda (o debería quedar) oculto al exterior, siendo el servidor web quien gestionase las peticiones.

Con dicha configuración el servidor web sirve los archivos estáticos mientras que solicita al servidor de aplicaciones la información derivada de la capa de dominio. En la figura 23 puede verse el diagrama de despliegue.

Módulo de modificación de contenido

Si desea prevenir la modificación del contenido del diccionario puede deshabilitar dicho módulo. Para ello basta con modificar el fichero `/WEB-INF/classes/resources/config.properties` actualizando la información `userModule = disabled` (recuerde que los cambios se aplicarán tras reiniciar el servidor Tomcat).

Gestión de usuarios

Actualmente el sistema no dispone de un sistema de gestión de usuarios. Para modificar la lista de usuarios del sistema deberá modificar directamente la tabla 'user' de la base de datos. Cuando modifique la información recuerde que el formato de almacenamiento de las contraseñas es un hash SHA-256. Existen utilidades online³⁰ para generarlo a partir de una palabra.

³⁰

Nota: Por motivos de seguridad NO se incluye una cuenta de usuario por defecto; para utilizar ciertas funcionalidades deberá crear una.

Online SHA-256 hash calculator: <http://www.xorbin.com/tools/sha256-hash-calculator>
hashgenerator.de: <http://www.hashgenerator.de/>

ANEXO III - Licencia GPL

Esta es una traducción no oficial de la Licencia Pública General GNU (GNU GPL) al español. No fue publicada por la Fundación para el Software Libre, y no establece legalmente los términos de distribución para software que utiliza la GNU GPL - sólo el texto original en inglés de la GNU GPL hace eso. De todas formas, esperamos que esta traducción ayude a los hispanohablantes a comprender mejor la GNU GPL.

LICENCIA PÚBLICA GENERAL GNU

Versión 3, 29 de junio de 2007

Copyright (C) 2007 Free Software Foundation, Inc. <<http://fsf.org/>>

Se permite la copia y distribución de copias literales de esta licencia, pero no está permitido modificarla.

Preámbulo

La Licencia Pública General GNU (GNU GPL) es una licencia libre, sin derechos para software y otro tipo de trabajos.

Las licencias para la mayoría del software y otros trabajos prácticos están destinadas a suprimir la libertad de compartir y modificar esos trabajos. Por el contrario, la Licencia Pública General GNU persigue garantizar su libertad para compartir y modificar todas las versiones de un programa—y asegurar que permanecerá como software libre para todos sus usuarios. Nosotros, La Fundación de Software Libre, usamos la Licencia Pública General GNU para la mayoría de nuestro software; y también se aplica a cualquier trabajo realizado de la misma forma por sus autores. Usted también puede aplicarla a sus programas.

Cuando hablamos de software libre, nos referimos a libertad, no a precio. Nuestras Licencias Públicas Generales están destinadas a garantizar la libertad de distribuir copias de software libre (y cobrar por ello si quiere), a recibir el código fuente o poder conseguirlo si así lo desea, a modificar el software o usar parte del mismo en nuevos programas libres, y a saber que puede hacer estas cosas.

Para proteger sus derechos, necesitamos evitar que otros le nieguen esos derechos o le pidan renunciar a ellos. Por lo tanto, usted tiene ciertas responsabilidades cuando distribuye copias del software, o si lo modifica: responsabilidades que persiguen respetar la libertad de otros.

Por ejemplo, si distribuye copias de tales programas, gratuitamente o no, debe transmitir a los destinatarios los mismos derechos que usted recibió. Debe asegurarse que ellos también reciban o puedan conseguir el código fuente. Y debe mostrarles estos términos y condiciones para que conozcan sus derechos.

Los desarrolladores que usen la GPL GNU protegen sus derechos de dos formas: (1) imponen derechos al software, y (2) le ofrecen esta Licencia para

que legalmente lo copie, distribuya y/o modifique.

Para proteger a desarrolladores y autores, la GPL expone claramente que no existe garantía alguna para este software libre. Para beneficio de ambos, usuarios y autores, la GPL establece que las versiones modificadas deberán estar identificadas como tales, para que cualquier problema no sea atribuido por error a los autores de versiones anteriores.

Algunos dispositivos están diseñados para negar al usuario la instalación o la ejecución de versiones modificadas del software que usan internamente, aunque el fabricante sí pueda hacerlo. Esto es completamente incompatible con el objetivo de proteger la libertad de los usuarios para modificar el software. Este tipo de abuso sistemático ocurre con productos de uso personal, que es precisamente donde es menos aceptable. Por tanto, hemos diseñado esta versión de la GPL para prohibir estas prácticas en esos productos. Si apareciesen problemas similares en otros ámbitos, estaremos preparados para extender estas prestaciones a las próximas versiones de la GPL, tanto como sea necesario para proteger la libertad de los usuarios.

Por último, todo programa está constantemente amenazado por las patentes de software. Los estados no deberían permitir que las patentes restrinjan el desarrollo y el uso de software en ordenadores de uso general; pero en aquellos que lo hagan, esperamos evitar el especial peligro que suponen las patentes, que aplicadas a un programa libre puedan hacerlo propietario en la práctica. Para prevenir eso, la GPL establece que las patentes no pueden usarse para convertir un programa en no-libre.

A continuación se exponen los términos y condiciones para la copia, distribución y modificación.

TÉRMINOS Y CONDICIONES

0. Definiciones.

En adelante "Esta Licencia" se refiere a la versión 3 de la Licencia Pública General GNU.

"Copyright" también significa "leyes similares al copyright" que son aplicables a otro tipo de trabajos, tales como las máscaras de semiconductores.

"El Programa" se refiere a cualquier trabajo con copyright al que se haya aplicado esta Licencia. Cada beneficiario es asimilable a "usted". "Beneficiarios" y "destinatarios" pueden ser personas físicas u organizaciones.

"Modificar" un trabajo significa copiar o adaptar todo o parte de un trabajo, exceptuando la copia exacta, de manera que se requiera permiso de copyright. El trabajo resultante se denomina "versión modificada" de un trabajo anterior o trabajo "basado en" el trabajo anterior.

Un "trabajo amparado" puede ser tanto el Programa no modificado como un trabajo basado en el Programa.

"Difundir" un trabajo significa hacer cualquier cosa con él, sin permiso, que le haga directa o indirectamente responsable de infringir leyes cubiertas por copyright, excepto la ejecución en un ordenador o la modificación de una copia

privada. La difusión incluye la copia, distribución (con o sin modificaciones), distribución pública, y en algunos países también otras actividades.

"Distribuir" un trabajo implica cualquier tipo de difusión que permite a la otra parte hacer o recibir copias. La mera interacción con un usuario mediante una red de ordenadores, sin transferir copia alguna, no se considera "distribución".

Una interfaz de usuario interactiva muestra "Avisos Legales Apropriados" siempre y cuando incluya características visuales apropiadas y destacadas que (1) muestren un aviso de copyright apropiado, y (2) indiquen al usuario que no existe garantía alguna para el trabajo (exceptuando las garantías que se hayan podido establecer), que los beneficiarios deben distribuir el trabajo según se establece en la presente Licencia, y cómo se puede ver una copia de esta Licencia. Si la interfaz muestra una lista de opciones o comandos, tales como menús, un elemento destacado en dicha lista cumple estos criterios.

1. Código Fuente.

El "código fuente" de un trabajo es el formato preferido para realizar modificaciones sobre él. "Código objeto" se refiere a cualquier formato del trabajo que no sea código fuente.

Una "Interfaz Estándar" se refiere a una interfaz que sea o bien un estándar oficial definido por una institución de estándares reconocida, o bien, en el caso de interfaces específicos para una determinado lenguaje de programación, una cuyo uso esté generalizada entre los desarrolladores que trabajan con ese lenguaje.

Las "Bibliotecas de Sistema" de un trabajo ejecutable incluyen a cualquier elemento, que no sea el trabajo completo, que (a) esté incluida/o de la misma forma que un componente principal, pero que no forme parte de ese componente principal, y (b) sólo sirva para habilitar la utilización del trabajo a través de ese componente principal, o para implementar un Interfaz Estándar para el cual está disponible una implementación pública en código fuente. Un "Componente Principal", en este contexto, se refiere a un componente principal y esencial (núcleo, sistema de ventanas y similares) del sistema operativo particular (en su caso) sobre el cual funcione el ejecutable, o un compilador utilizado para generar el trabajo, o un intérprete del código objeto utilizado para ejecutarlo.

La "Fuente Correspondiente" de un trabajo en código objeto se refiere a todo código fuente necesario para generar, instalar, y (en el caso de trabajos ejecutables) ejecutar el código objeto y modificar el trabajo, incluyendo guiones que controlen esas actividades. Sin embargo, no se incluyen las Bibliotecas de Sistema del trabajo, o herramientas de propósito general o programas gratuitos habitualmente disponibles y usados sin ninguna modificación para realizar estas actividades pero que no forman parte del trabajo. Por ejemplo, la Fuente Correspondiente incluye los archivos de definición de interfaz asociados con archivos fuente del trabajo, y el código fuente de las bibliotecas compartidas o subprogramas enlazados dinámicamente que el programa requiere por diseño, como la comunicación de datos intrínseca o el control de flujo entre esos subprogramas y otras partes del trabajo.

La Fuente Correspondiente no incluye necesariamente aquello que los usuarios pueden regenerar automáticamente a partir de otras partes de la Fuente Correspondiente.

La Fuente Correspondiente de un trabajo en código fuente es ese mismo trabajo.

2. Permisos Básicos.

Todos los derechos garantizados por esta Licencia se otorgan como copyright del Programa, y se proporcionan de manera irrevocable siempre y cuando se cumplan las condiciones establecidas. Esta Licencia afirma explícitamente su permiso ilimitado para ejecutar el Programa sin modificaciones. El resultado de la ejecución de un programa amparado está cubierto por esta Licencia sólo en el caso de que la salida, por su contenido, constituya un trabajo amparado. Esta Licencia reconoce sus derechos de uso razonable u otro equivalente, tal y como determina la ley de copyright.

Usted podrá realizar, ejecutar y difundir trabajos amparados que no distribuya, sin condición alguna, siempre y cuando no tenga otra licencia más restrictiva. Podrá distribuir trabajos amparados a terceros con el mero objetivo de que ellos hagan modificaciones exclusivamente para usted, o para que le proporcionen ayuda para ejecutar esos trabajos, siempre que cumpla los términos de esta Licencia distribuyendo todo el material de cuyo copyright no posee el control. Aquellos que realicen o ejecuten los trabajos amparados para usted deben hacerlo exclusivamente en su nombre, bajo su dirección y control, con términos que les prohíban realizar copias de su material con copyright al margen de la relación con usted.

La distribución bajo otras circunstancias se permite únicamente bajo las condiciones establecidas más abajo. No está permitido sublicenciar; la cláusula 10 lo hace innecesario.

3. Protección de Derechos Legales de los Usuarios frente a Leyes Anti-Burla.

Ningún trabajo amparado debe considerarse parte de una medida tecnológica efectiva, a tenor de lo establecido en cualquier ley aplicable que cumpla las obligaciones expresas en el artículo 11 del tratado de copyright WIPO adoptado el 20 de diciembre de 1996, o leyes similares que prohíben o restringen la burla de tales medidas.

Cuando distribuya un trabajo amparado, renuncia a cualquier poder legal para prohibir la burla de medidas tecnológicas mientras tales burlas se realicen en ejercicio de derechos amparados por esta Licencia respecto al trabajo amparado; además, usted negará cualquier intención de limitar el uso o modificación del trabajo con el objetivo de imponer, al trabajo de los usuarios, sus derechos legales o de terceros para prohibir la burla de medidas tecnológicas.

4. Distribución de copias literales.

Usted podrá distribuir copias literales del código fuente del Programa tal y como lo ha recibido , por cualquier medio, siempre que publique de forma clara y llamativa en cada copia el correspondiente aviso de copyright ; mantenga intactos todos los avisos que establezcan que esta Licencia y cualquier término no-permisivo añadido y acorde con la cláusula 7 son aplicables al código; mantenga intactos todos los avisos de ausencia de garantía; y proporcione a todos los destinatarios una copia de esta Licencia junto con el Programa.

Usted podrá cobrar cualquier importe o no cobrar nada por cada copia que distribuya, y podrá ofrecer soporte o protección de garantía mediante un pago.

5. Distribución de Versiones Modificadas de Código.

Usted podrá distribuir un trabajo basado en el Programa, o las modificaciones que lo producen a partir del Programa, como código fuente en virtud de los términos establecidos en la cláusula 4, siempre que cumpla todas las condiciones siguientes:

- (a) El trabajo debe incluir avisos destacados indicando que usted lo ha modificado y dando una fecha pertinente.
- (b) El trabajo debe incluir avisos destacados indicando que está realizado a tenor de lo dispuesto en la presente Licencia y en cualquier otra condición añadida en virtud de la cláusula 7. Este requisito modifica el requisito de "mantener intactos todos los avisos" expuesto en la cláusula 4.
- (c) En virtud del presente documento, usted deberá aplicar la licencia al trabajo completo, como un todo, a cualquier persona que esté en posesión de una copia. Por lo tanto, esta Licencia se aplicará junto con cualquier otra condición adicional aplicable de la cláusula 7, al conjunto completo del trabajo y todas y cada una de sus partes, independientemente de como sean agrupadas o empaquetadas. Esta Licencia no permite ser aplicada al trabajo de ninguna otra forma, pero no se anula dicho permiso si usted lo ha recibido por separado.
- (d) Si el trabajo tiene interfaces de usuario interactivos, cada uno debe mostrar Avisos Legales Apropriados; sin embargo, si el Programa tiene interfaces interactivos que no muestran Avisos Legales Apropriados, su trabajo no tiene porqué modificarlos para que lo hagan.

Un conjunto o recopilación formado por un trabajo amparado y otros trabajos distintos e independientes, que por su naturaleza no sean ampliaciones del trabajo amparado, que no se combinen con él de alguna forma para dar lugar a un programa mayor, y que estén ubicados en un medio de distribución o almacenamiento, se denomina "paquete" si la recopilación y su copyright al completo no son usados para limitar el acceso o los derechos legales de los usuarios de la recopilación, más allá de lo que permita el trabajo individual. La inclusión de

un trabajo amparado en un paquete no hace aplicable esta Licencia al resto de elementos del paquete.

6. Distribución de código No-fuente.

Usted podrá distribuir el código objeto de un trabajo amparado en virtud de los términos de las cláusulas 4 y 5, siempre que también distribuya las Fuentes Correspondientes en código máquina, de acuerdo con los términos establecidos en esta Licencia, de alguna de las siguientes maneras:

- (a) Distribuir el código objeto en, o embebido en, un producto físico (incluyendo medios de distribución físicos), acompañado de las Fuentes Correspondientes en un medio físico duradero y que sea utilizado habitualmente para el intercambio de software.
- (b) Distribuir el código objeto en, o embebido en, un producto físico (incluyendo medios de distribución físicos), acompañado de una oferta por escrito, válida al menos durante tres años y válida durante el tiempo en el que usted ofrezca recambios o soporte para ese modelo de producto, con el fin de ofrecer al poseedor del código objeto (1) una copia de las Fuentes Correspondientes a todo el software del producto que esté cubierto por esta Licencia, en un medio físico duradero habitual para el intercambio de software, a un precio no mayor que su coste razonable por distribuir físicamente las fuentes, o (2) acceso para copiar las fuentes correspondientes desde un servidor de red sin coste alguno.
- (c) Distribuir copias individuales del código objeto junto con una copia de la oferta por escrito para/con el fin de proporcionar las Fuentes Correspondientes. Esta alternativa sólo está permitida ocasionalmente, pero no de forma comercial, y solamente si usted recibió el código objeto junto con una oferta parecida, de acuerdo con la subcláusula 6b.
- (d) Distribuir el código objeto ofreciendo acceso desde un lugar determinado (gratuitamente o mediante pago), y ofrecer acceso equivalente a las Fuentes Correspondientes de la misma forma y en el mismo lugar sin cargo añadido. No es necesario exigir a los destinatarios que copien las Fuentes Correspondientes junto con el código objeto. Si el lugar para copiar el código objeto es un servidor de red, las Fuentes Correspondientes pueden estar en un servidor diferente (gestionado por usted o terceros) que ofrezca facilidades de copia equivalentes, siempre que mantenga instrucciones claras junto al código objeto especificando dónde encontrar las Fuentes Correspondientes. Independientemente de qué servidores alberguen las Fuentes Correspondientes, usted seguirá obligado a asegurar que estarán disponibles durante el tiempo necesario para cumplir estos requisitos.
- (e) Distribuir el código mediante transferencias entre usuarios, siempre que informe a otros usuarios dónde se ofrecen el código objeto y las Fuen-

tes Correspondientes de forma pública sin cargo alguno, tal y como se establece en la subcláusula 6d.

Una parte separable del código objeto, cuyo código fuente esté excluido de las Fuentes Correspondientes como Biblioteca de Sistema, no necesita ser incluida en la distribución del código objeto del trabajo.

Un "Producto de Usuario" es tanto (1) un "producto de consumo", que se refiere a cualquier propiedad personal tangible habitualmente utilizada para fines personales, familiares o domésticos, o (2) cualquier cosa diseñada o vendida para ser incorporada como extensión/expansión para otro producto. Para determinar si un producto es un producto de consumo, los casos dudosos se resolverán favoreciendo el amparo. En el caso de un producto concreto recibido por un usuario particular, "de uso habitual" se refiere al uso típico o corriente de ese tipo de producto, independientemente de la situación del usuario particular o de la forma en que el usuario concreto utilice, o pretenda o se espere que pretenda utilizar, el producto. Un producto es un producto de consumo independientemente de si el producto tiene usos sustancialmente comerciales, industriales o distintos del consumo, a menos que tales usos representen la única forma posible de utilizar el producto.

Las "Instrucciones de Instalación" para un Producto de Usuario se refieren a cualquier método, procedimiento, clave de autorización, u otro tipo de información necesaria para instalar y ejecutar una versión modificada de un trabajo amparado en ese Producto de Usuario a partir de una versión modificada de las Fuentes Correspondientes. Las instrucciones deben ser suficientes para asegurar el funcionamiento continuo del código objeto modificado sin ningún tipo de condicionamiento o intromisión por el simple hecho de haber sido modificado.

Si, bajo las premisas de esta cláusula, usted distribuye el código objeto de un trabajo en, o con un Producto de Usuario o específicamente para ser usado en el mismo, y la distribución forma parte de una transacción donde los derechos de posesión y uso del Producto de Usuario se transfieren al destinatario a perpetuidad o durante un plazo fijo de tiempo (independientemente de las características de la transacción), las Fuentes Correspondientes distribuidas bajo estos supuestos deben acompañarse de las Instrucciones de Instalación. Sin embargo, estos requerimientos no se aplican si ni usted ni terceros tienen posibilidad de instalar el código objeto modificado en el Producto de Usuario (por ejemplo, el trabajo ha sido instalado en memoria de sólo lectura, ROM):

El requerimiento de proporcionar Información de Instalación no incluye el hecho de continuar proporcionando servicio de soporte, garantía, o actualizaciones para un trabajo que haya sido modificado o instalado por el destinatario, o para el Producto de Usuario en el que se haya modificado o instalado. El acceso a la red puede ser denegado cuando la propia modificación afecte materialmente y de forma adversa a la operación de la red o viole las reglas y protocolos de comunicación en la red.

Las Fuentes Correspondientes distribuidas, y las Instrucciones de Instalación proporcionadas de acuerdo con esta cláusula, deben figurar en un formato documentado públicamente (y con una implementación disponible para el pú-

blico en código fuente), y no deben necesitar claves de acceso especiales para la descompresión, lectura o copia.

7. Condiciones adicionales.

Los "Permisos Adicionales" son condicionantes que amplían los términos de esta Licencia permitiendo excepciones a una o más de sus condiciones. Los Permisos Adicionales que son aplicables al Programa completo deberán ser tratados como si estuviesen incluidos en esta Licencia, hasta los límites de validez impuestos por las leyes aplicables. Si los permisos adicionales se aplicasen sólo a una parte del Programa, esa parte podría ser usada de forma independiente en virtud de dichos permisos, pero el Programa completo seguiría estando afectado por esta Licencia con independencia de los permisos adicionales.

Cuando distribuya una copia de un trabajo amparado, usted podrá opcionalmente eliminar cualquier permiso adicional de esa copia, o de alguna parte del mismo. (Los permisos adicionales pueden haber establecido que sea requerida su eliminación en ciertos supuestos si usted modifica el trabajo.) Usted puede establecer permisos adicionales en material añadido por usted a un trabajo amparado, sobre el cual tiene o podrá aportar sus permisos de copyright correspondientes.

Sin contravenir cualquier otra estipulación en esta Licencia, usted podrá, para el material que añada a un trabajo amparado, (si está autorizado por los poseedores de copyright de ese material) añadir condiciones a esta Licencia con los siguientes términos:

- (a) Ausencia de garantía o limitación de responsabilidad diferente de los términos establecidos en las cláusulas 15 y 16 de esta Licencia; u
- (b) Obligación de mantener determinados avisos legales razonables o atribuciones de autoría en el material o en los Avisos Legales Correspondientes mostrados por los trabajos que lo contengan; o
- (c) Prohibir la tergiversación del origen del material, o solicitar que las diferencias respecto a la versión original sean señaladas de forma apropiada en las versiones modificadas del material; o
- (d) Limitar la utilización de los nombres de los autores o beneficiarios del material con fines divulgativos; o
- (e) Negarse a ofrecer derechos afectados por leyes de registro para el uso de marcas empresariales, registradas o de servicio; o
- (f) Exigir indemnización a los autores y poseedores de la licencia de ese material, por parte de cualquier persona que distribuya el material (o versiones modificadas del mismo), estableciendo obligaciones contractuales de responsabilidad sobre el destinatario, para cualquier responsabilidad que estas obligaciones contractuales impongan directamente sobre los autores y poseedores de licencia.

Cualesquiera otras condiciones adicionales no-permisivas son consideradas "otras restricciones" en el contexto de la cláusula 10. Si el Programa, tal cual lo recibió, o cualquier parte del mismo, contiene un aviso indicando que está amparado por esta Licencia junto a una cláusula de restricción posterior específica, usted podrá suprimir esa cláusula. Si un documento de licencia contiene una restricción de este tipo pero permite modificar la licencia o la distribución en virtud de la presente Licencia, usted podrá añadirla al material de un trabajo amparado por los términos de ese documento de licencia, siempre que dicha restricción no se mantenga tras la modificación de la licencia o la distribución.

Si añade condiciones para un trabajo amparado, a tenor de lo establecido en la presente cláusula, usted deberá ubicar, en los archivos fuente involucrados, una declaración de los términos adicionales aplicables a esos archivos, o un aviso indicando dónde localizar los términos aplicables.

Las condiciones adicionales, permisivas o no, deben aparecer por escrito como licencias separadas, o figurar como excepciones; de todas formas, los requisitos anteriores siempre son aplicables.

8. Cancelación.

Usted no podrá distribuir o modificar un trabajo amparado salvo de la forma en la que se ha previsto expresamente en esta Licencia. Cualquier intento diferente de distribución o modificación será considerado nulo, y automáticamente cancelará sus derechos respecto a esta Licencia (incluyendo cualquier patente conseguida según el párrafo tercero de la cláusula 11).

Sin embargo, si deja de violar esta Licencia, entonces su licencia desde el poseedor del copyright correspondiente será restituida (a) provisionalmente, a menos que y hasta que el poseedor del copyright dé por terminada explícita y permanentemente su licencia, y (b) permanentemente, si el poseedor del copyright no le ha notificado por algún cauce de la violación no después de los 60 días posteriores al cese.

Además, su licencia desde el poseedor del copyright correspondiente será restituida permanentemente si el poseedor del copyright le notifica de la violación por algún cauce, es la primera vez que recibe la notificación de violación de esta Licencia (para cualquier trabajo) de ese poseedor de copyright, y usted subsana la violación antes de 30 días desde la recepción del aviso.

La cancelación de sus derechos según esta cláusula no da por canceladas las licencias de terceros que hayan recibido copias o derechos a través de usted con esta Licencia. Si sus derechos han finalizado y no han sido restituidos de forma permanente, usted no está capacitado para recibir nuevas licencias para el mismo material en virtud de la cláusula 10.

9. Aceptación no obligatoria por tenencia de copias.

No está obligado a aceptar esta Licencia por recibir o ejecutar una copia del Programa. La distribución de un trabajo amparado surgida simplemente como consecuencia de la transmisión entre usuarios para obtener una copia tampoco

requiere aceptación. Sin embargo, únicamente esta Licencia le otorga permiso para distribuir o modificar cualquier trabajo amparado. Estas acciones infringen el copyright si usted no acepta las los términos y condiciones de esta Licencia. Por lo tanto, al modificar o distribuir un trabajo amparado, usted indica que acepta la Licencia.

10. Herencia automática de licencia para destinatarios.

Cada vez que distribuya un trabajo amparado, el destinatario recibirá automáticamente una licencia desde los poseedores originales, para ejecutar, modificar y distribuir ese trabajo, al amparo de los términos de esta Licencia. Usted no será responsable de asegurar el cumplimiento por terceros de esta Licencia.

Una "transacción de entidad" es una transacción que transfiere el control de una organización, o todos los bienes sustanciales de una, o subdivide una organización, o fusiona organizaciones. Si la distribución de un trabajo amparado surge de una transacción de entidad, cada parte involucrada en esa transacción que reciba una copia del trabajo, también recibe todas y cada una de las licencias existentes del trabajo que la parte interesada tuviese o pudiese ofrecer según el párrafo anterior, además del derecho a tomar posesión de las Fuentes Correspondientes del trabajo a través de la parte interesada, si está en poder de dicha parte o se puede conseguir con un esfuerzo razonable.

Usted no podrá imponer restricciones posteriores en el ejercicio de los derechos otorgados o concedidos en virtud de la presente Licencia. Por ejemplo, usted no puede imponer a la licencia pagos, derechos u otros cargos por el ejercicio de los derechos otorgados según esta Licencia; además no podrá iniciar litigios (incluyendo demandas o contrademandas en pleitos) alegando que se infringen patentes por cambiar, usar, vender, ofrecer en venta o importar el Programa, o cualquier parte del mismo.

11. Patentes.

Un "colaborador" es un poseedor de copyright que autoriza el uso del Programa o un trabajo en el que se base el Programa bajo los términos y condiciones establecidos en la presente Licencia. El trabajo con esta licencia se denomina "versión en colaboración" con el colaborador.

Todas las reivindicaciones de patentes en posesión o controladas por el colaborador se denominan "demandas de patente original", ya sean existentes o adquiridas con posterioridad, que hayan sido infringidas de alguna forma permitida por esta Licencia, al hacer, usar o vender la versión en colaboración, pero sin incluir demandas que sólo sean infracciones como consecuencia de modificaciones posteriores de la versión en colaboración. Para aclarar esta definición, "control" incluye el derecho de conceder sublicencias de patente de forma que no contravenga los requisitos establecidos en la presente Licencia.

Cada colaborador le concede a usted una licencia de la patente no-exclusiva, global y libre de derechos bajo las reivindicaciones de patente de origen del colaborador, para el uso, modificación, venta, ofertas de venta, importación y

otras formas de ejecución, modificación y redistribución del contenido de la versión en colaboración.

En los siguientes tres párrafos, una "licencia de patente" se refiere a cualquier acuerdo o compromiso expreso y manifiesto, cualquiera que sea su denominación, que no imponga una patente (como puede ser el permiso expreso para ejecutar una patente o acuerdos para no imponer demandas por infracción de patente). "Conceder" estas licencias de patente a un tercero significa llegar a tal tipo de acuerdo o compromiso que no imponga una patente al tercero.

Si usted distribuye un trabajo amparado, conociendo que está afectado por una licencia de patente, y no están disponibles de forma pública para su copia las Fuentes Correspondientes, sin cargo alguno y bajo los términos de esta Licencia, ya sea a través de un servidor de red público o mediante cualquier otro medio, entonces usted deberá o bien (1) permitir que sean públicas las Fuentes Correspondientes, o (2) tratar de eliminar los beneficios de la licencia de patente para este trabajo en particular, o (3) tratar de extender, de una forma que no contravenga los requisitos de esta Licencia, la licencia de patente a terceros. "Conocer que está afectado" significa que usted tiene conocimiento real de que, para la licencia de patente, la distribución del trabajo amparado en un determinado país, o el uso del trabajo amparado por sus destinatarios en un determinado país, infringiría una o más patentes existentes en ese país que usted considera aplicables por algún motivo.

Si, de conformidad con alguna transacción o acuerdo(o en un proceso relacionado con ellos), usted distribuye o distribuye con fines de distribución, un trabajo amparado, concediendo una licencia de patente para algún tercero que reciba el trabajo amparado, y autorizándole a usar, distribuir, modificar o distribuir una copia específica del trabajo amparado, entonces la licencia de patente que usted otorgue se extiende automáticamente a todos los receptores del trabajo amparado y cualquier trabajo basado en el mismo.

Una licencia de patente es "discriminatoria" si no incluye dentro de su ámbito de cobertura, prohíbe el ejercicio, o está condicionada a no ejercitar uno o más de los derechos que están específicamente otorgados por esta Licencia. Usted no debe distribuir un trabajo amparado si está implicado en un acuerdo con terceros que estén relacionados con el negocio de la distribución de software, en el que usted haga pagos relacionados con su actividad de distribución del trabajo, y donde se otorgue, a cualquier receptor del trabajo amparado, una licencia de patente discriminatoria (a) en relación con las copias del trabajo amparado distribuido por usted (o copias hechas a partir de éstas), o (b) directa o indirectamente relacionadas con productos específicos o paquetes que contengan el trabajo amparado, a menos que usted forme parte del acuerdo, o que esa licencia de patente fuese otorgada antes del 28 de marzo de 2007.

Ninguna disposición de esta Licencia se considerará como excluyente o limitante de la aplicación de cualquier otra licencia o defensas legales contra la violación de las leyes de propiedad intelectual a que pudiera tener derecho bajo la ley de propiedad intelectual vigente.

12. No condicionamiento de la libertad de terceros.

Si a usted le son impuestas condiciones que contravienen las estipuladas en la presente Licencia (ya sea por orden judicial, acuerdo u otros), no quedará eximido de cumplir las condiciones de esta Licencia. Si usted no puede distribuir un trabajo amparado cumpliendo simultáneamente sus obligaciones con esta Licencia y con cualquier otra pertinente, entonces no podrá distribuirlo de ninguna forma. Por ejemplo, si usted se compromete con términos que le obligan a obtener derechos por la distribución a terceros, la única forma de satisfacer ambos condicionantes y esta Licencia es abstenerse completamente de distribuir el Programa.

13. Uso conjunto con la Licencia Pública General Affero GNU.

Sin contravenir las disposiciones de la presente Licencia, usted tendrá permiso para enlazar o combinar cualquier trabajo amparado con otro trabajo amparado por la versión 3 de la Licencia Pública General Affero GNU y formar un solo trabajo combinado, y distribuir el trabajo resultante. Los términos de esta Licencia seguirán siendo aplicables a la parte formada por el trabajo amparado, pero los condicionantes especiales de la Licencia Pública General Affero GNU, en su cláusula 13, relativos a la interacción mediante redes, serán aplicables a la combinación de ambas partes.

14. Versiones Revisadas de esta Licencia.

La Fundación para el Software Libre podrá publicar revisiones y/o nuevas versiones de la Licencia Pública General GNU de vez en cuando. Esas versiones serán similares en espíritu a la versión actual, pero podrán diferir en algunos detalles para afrontar nuevos problemas o situaciones.

A cada versión se le da un número distintivo. Si el Programa especifica que le es aplicable cierto número de versión de la Licencia Pública General o "cualquier versión posterior", usted tendrá la posibilidad de adoptar los términos y condiciones de la versión indicada o de cualquier otra versión posterior publicada por la Fundación para el Software Libre. Si el Programa no especifica un número de versión de la Licencia Pública General, usted podrá elegir cualquier versión que haya sido publicada por la Fundación para el Software Libre.

Si el Programa especifica que un apoderado/representante puede decidir qué versiones de la Licencia Pública General pueden aplicarse en el futuro, la declaración pública de aceptación que el apoderado/representante haga de una versión le autoriza a usted con carácter permanente a elegir esa versión para el Programa.

Versiones posteriores de la licencia podrán otorgarle permisos adicionales o diferentes. Sin embargo, no podrán imponerse obligaciones adicionales a cualquier autor o poseedor de copyright como consecuencia de que usted adopte una versión posterior.

15. Ausencia de Garantía.

EL PROGRAMA NO TIENE GARANTÍA ALGUNA, HASTA LOS LÍMITES PERMITIDOS POR LAS LEYES APLICABLES. SALVO CUANDO SE ESTABLEZCA LO CONTRARIO POR ESCRITO, EL POSEEDOR DEL COPYRIGHT Y/O TERCEROS PROPORCIONARÁN EL PROGRAMA "TAL CUAL" SIN GARANTÍA DE NINGÚN TIPO, YA SEA EXPLÍCITA O IMPLÍCITA, INCLUYENDO, PERO SIN LIMITARSE A, LAS GARANTÍAS IMPLÍCITAS MERCANTILES Y DE APTITUD PARA UN PROPÓSITO DETERMINADO. USTED ASUMIRÁ CUALQUIER RIESGO RELATIVO A LA CALIDAD Y RENDIMIENTO DEL PROGRAMA. SI EL PROGRAMA FUESE DEFECTUOSO, USTED ASUMIRÁ CUALQUIER COSTE DE SERVICIO, REPARACIÓN O CORRECCIÓN.

16. Limitación de Responsabilidad.

EN NINGÚN CASO, SALVO REQUERIMIENTO POR LEYES APLICABLES O MEDIANTE ACUERDO POR ESCRITO, PODRÁ UN POSEEDOR DE COPYRIGHT, O UN TERCERO QUE MODIFIQUE O DISTRIBUYA EL PROGRAMA SEGÚN LO INDICADO ANTERIORMENTE, HACERLE A USTED RESPONSABLE DE DAÑO ALGUNO, INCLUYENDO CUALQUIER DAÑO GENERAL, ESPECIAL, OCASIONAL O DERIVADO QUE SURJA DEL USO O LA INCAPACIDAD DE USO DEL PROGRAMA (INCLUYENDO PERO SIN LIMITARSE A LA PÉRDIDA DE DATOS O LA PRESENTACIÓN NO PRECISA DE LOS MISMOS O A PÉRDIDAS SUFRIDAS POR USTED O TERCEROS O AL FALLO DEL PROGRAMA AL INTERACTUAR CON OTROS PROGRAMAS), INCLUSO EN EL CASO DE QUE EL POSEEDOR O UN TERCERO HAYA SIDO ADVERTIDO DE LA POSIBILIDAD DE TALES DAÑOS.

17. Interpretación de las cláusulas 15 y 16.

Si la ausencia de garantía y la limitación de responsabilidad descrita anteriormente no tuviesen efecto legal a nivel local en todos sus términos, los juzgados aplicarán las leyes locales que más se aproximen a la exención de responsabilidad civil en lo relativo al Programa, a menos que la copia del Programa esté acompañada mediante pago de una garantía o compromiso de responsabilidad.

FIN DE TÉRMINOS Y CONDICIONES